(1)

AFIT/GOR/ENS/93M-02

USING ANIMATION IN THE VALIDATION OF SIMULATION MODELS

THESIS

Michael L. Carpenter
Captain, USAF

AFIT/GOR/ENS/93M-02

Approved for public release; distribution unlimited

AFIT/GOR/ENS/93M-02

USING ANIMATION IN THE VALIDATION OF SIMULATION MODELS

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Operations Research

Michael L. Carpenter, B.S.

Captain, USAF

March, 1993

THESIS APPROVAL

STUDENT: Capt Michael L. Carpenter **CLASS:** GOR-93M

THESIS TITLE: Using Animation in the Validation of Simulation Models

DEFENSE DATE: February 8, 1993

COMMITTEE:	NAME/DEPARTMENT	SIGNATURE
Advisor	Lt Col Kenneth W. Bauer, Jr./ENS	Kenneth W Bauer Jr.
Reader	Col Thomas F. Schuppe/LA	Thomas F Schuppe

Acknowledgments

This thesis is the result of the help, support, and encouragement of many individuals. My first thank you goes to Lt Col Kenneth Bauer, my thesis advisor, and Col Thomas Schuppe, my reader. Their guidance was particularly helpful in nailing down a topic and keeping my writing in focus. I would also like to thank Lt Col Larry Wolfe of the Air Force Operational Test and Evaluation Center (AFOTEC) for his support by providing ideas and software. Special thanks goes to Dr. Michael Vidulich of Wright-Patterson's Armstrong Laboratory for his help in designing the experiment and proofing the results. Also, I extend my appreciation to the individuals who volunteered to participate in the experiment. This project would not have been possible without them.

In conclusion, I offer my thanks and love to my wife and children. Their encouragement and understanding meant more than I could ever express. In the end, though, the glory and thanks go to my Lord Jesus Christ. This thesis has been lifted up in prayer since my first day at AFIT. His guidance led me to the right topic, advisor, and reader, and He has watched over every step of this process. Truly, I can do nothing apart from Him (John 15:6).

Michael L. Carpenter

Table of Contents

Page

List of Figures

Figure		Page

List of Tables

AFIT/GOR/ENS/93M-02

Abstract

This study examined three aspects of animation (movement, color, and detail of icons) to determine which one (or ones) best communicated the operation of a simulation model. The procedure was done in the context of using animation to establish a model's face validity. Movement, color, and detail of icons were looked at individually and in combination. The ability to communicate was measured both subjectively and objectively. The subjective measures were a selection of "best" and "worst" animation types where "best" and "worst" referred to how well an animation communicated, and a pairwise comparison of the animation types which resulted in preference ratings for each animation. There were seven different scenarios containing various problems with the system. The objective measures were subject problem identification accuracy and time delay of problem identification. The results showed that movement in animations was always preferred to a lack of movement in animations. However, movement, color, and detail of icons in combination was preferred the most. Objectively, movement was the most important aspect. The subjects performed equally well for all the animations with movement and, when there was no movement, the subjects' performance dropped equally.

USING ANIMATION IN THE VALIDATION OF SIMULATION MODELS

I. Introduction

Background

Over the past two decades simulation modeling in the military has increased significantly. Today the military uses simulation models in many areas: defense planning, weapon system evaluation, logistics, and the like. Models have become essential in military analysis because a valid simulation model can point out potential trouble spots or potential solutions in a system without the expense in time and money of operating the actual system. Also, simulation models are valuable tools for planning and evaluating future systems.

In general, models are abstractions of reality that attempt to capture the essence of the problems, events, or actions being considered. Thus, no effort is made to include every detail in a model. In the first place that would be impossible, and secondly, including every detail is unnecessary. Developing a model is like an artist painting a portrait. The artist initially draws a sketch of what is to be painted. The sketch, like the model, contains the essentials but omits the details. Just as the sketch cannot be a substitute for the portrait, a model cannot be a substitute for reality. However, a sketch well done leads to a better portrait, and an accurate model leads to a better understanding of the literal or proposed situation. Models are developed in various forms (e.g. mathematical, verbal, or pictorial). Simulation models are generally considered to be models that are developed as computer programs.

Although simulation models are used extensively in decision making and problem solving, many decision makers lack confidence in simulation model results (Sargent, 1991:37). If models are to be legitimate decision aids, it is critical that they be validated. Simulation model validation is "substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model" (Schlesinger and others, 1979:103). The validation process consists of performing tests and evaluations during the development of a simulation model. These tests range from a review of model assumptions to detailed statistical procedures. Although model verification is closely linked with model validation, the two should not be confused. Simulation model verification is the process of ensuring that the actual computer program is error free. As Whitner and Balci state, "Validation deals with building the right model, verification deals with building the model right" (Whitner and Balci, 1989:559).

Even though much has been written concerning simulation model validation, and many validation techniques have been developed, no simulation model can ever be fully validated. A successful result from a validation procedure can only increase the validity of a model. Thus, the modeler desires to tip the scales of validation towards complete validity as far as possible, all the while knowing that there will always be some uncertainty weighing down the other side.

Computer animation is one of many techniques used in the process of simulation model validation (and verification). Through computer animation a "model's operational behavior is displayed graphically as the model moves through time" (Sargent, 1991:39). Animation is becoming more popular because animation software has dropped in price and increased in quality. Increased computer graphics capability allows the modeler to see the simulated operation of the system instead of just a printout of statistics. In addition, animation can be used to enhance a model's credibility. A credible model is a model the decision makers are willing to use because they have confidence in the model's results (Sargent, 1991:37). According to Law and Kelton, the ability to increase a model's credibility is the main reason for animation's expanding use (Law and Kelton, 1991:241). Thus, the graphical display of a model through animation can add credence to a model and increase the confidence in a model.

Problem

The use of computer animation in simulation model validation is increasing; however, there are few guidelines for using animation in conjunction with accepted simulation validation techniques. One primary method used in model validation is face validity. "Face validity is asking people knowledgeable about the system whether the model and/or its behavior is reasonable" (Sargent, 1991:39). In order to establish the face validity of a model, the assumptions and operation of the model must be communicated to the system expert. Animation is one tool for accomplishing that. Thus, this research focused on determining which aspects of animation are the most useful for communicating the operation of the model.

Assumptions and Scope

Simulation Model. The simulation model used for this research was taken from a SLAM (Simulation Language for Alternative Modeling) textbook. A simple model was chosen so that some of the basics of animation could be examined without the problems associated with a complex model. The simulation models one bulldozer, four trucks, and two man-machine loaders. The bulldozer stockpiles material for the loaders. The loaders place the material into the trucks. Once a truck is loaded it hauls the material away, dumps the load, and then returns for another load. The Truck Hauling Situation model will hereafter be referred to as the Loader model (Pritsker, 1986:237-242).

Animation Software. The model was animated with Proof Animation. Proof Animation (hereafter referred to as Proof) is a PC-based, "post-processing" animation software. Post-processing (or playback) means the animation is seen after the simulation is run. The events or state changes are recorded in a file during the simulation run and then "played back" by the animator. This is in contrast to "concurrent" animation. Concurrent means the animation is seen while the simulation is running (Law and Kelton, 1991:241). There are advantages and disadvantages to each type of animation. Concurrent animation usually allows the analyst or user to immediately see the result of changes they make while the simulation is running; however, certain changes cannot be made while the simulation is running. Also, the end (or any other part) of a simulation run can only be seen after viewing all that comes before. That is, "fast-forwarding" is not possible. Post-processing, on the other hand, does allow one to view to any part of a simulation run without waiting through the previous portions of the simulation, but changes cannot be made and seen interactively (Brunner and others, 1991:91).

Scope of Animation. Three aspects of animation were considered:

- Movement
- Detail of Icons
- Color

Each of the above was looked at individually and in combination. Other factors that could be considered include graphs, perspective (as in two-dimensional or three-dimensional), concurrent animation versus playback, and speed of animation.

Approach

The main reason for using animation is that it improves communication. Thus, the procedure was to animate the Loader model, and show the animations to volunteers. They determined how useful the animations were in enhancing face validity and which animation type was most beneficial. This was measured in two ways. First, each animation showed a different scenario. The subject attempted to determine if there were any efficiency problems with the system modeled (queue buildups, long idle times, and the like). The time delay in identifying a potential problem and the identification error rate were recorded. Second, after viewing the animations, subjects performed a pairwise comparison of each animation's ability to communicate. A rating for each animation was calculated from the pairwise comparisons. The pairwise comparison procedure is called the Analytic Hierarchy Process and is discussed in the next chapter.

Overview

The next chapter reviews the current literature concerning simulation model validation and animation. Also, a section discussing the Analytic Hierarchy Process is included. Chapter 3 gives a detailed description of the approach to the research, the Loader model, the animations, the experimental design and data collection, and the analysis methods used. The fourth chapter is a discussion of the results, and the final chapter contains conclusions and recommendations. The data collection forms are shown in Appendix A. Appendix B is the SAS output used in the analysis, and Appendix C contains the SLAM and Fortran code.

II. Literature Review

Scope and Method of Presentation

This review will cover information about validation of simulation models, the use of animation in validation, and the Analytic Hierarchy Process (AHP). The discussion will begin with model validation in general and include animation's applicability to validation. A review of information dealing with the use of animation in validating models will follow, and a look at the AHP will conclude the discussion.

Discussion

Model Validation. Since the 1960's much has been written concerning model validation. Balci and Sargent list over 300 articles in a comprehensive bibliography (through 1983) of articles dealing with validation and credibility assessment of models (Balci and Sargent, 1984:15-27). Carson defines validation and a credible model as follows:

- Validation: "The process whereby the modeler and end user ask the questions: How accurately does the model represent reality? Can the model be used in place of the real system for the purpose of making decisions concerning the real system" (Carson, 1989:552)?
- Credible Model: A model "that is accepted by the client as being sufficiently accurate to be used as an aid in making decisions" (Carson, 1989:552).

One of the most published authors in the area of validation is Sargent. He emphasizes the relationship between the purpose of a model and the validity of a model. A model's validity can only be determined in relation to the purpose of a model. If a model is designed to evaluate several different problems, it must be validated for each of the problems considered. Also, a model can never be considered completely valid. He states, "It is often too costly and time consuming to determine that a model is absolutely valid over the complete domain of its intended applicability" (Sargent, 1991:37). There are only degrees or levels of validity (Sargent, 1991:37).

In his discussion of validation techniques Sargent distinguishes between subjective and objective (statistical) validation techniques. Sargent lists 15 subjective techniques, one being animation. He states that the model validation process is included in the process of model development and is concerned with four different evaluations: conceptual model validity, computerized model verification, operational validity, and data validity. All of the validation techniques he lists are useful in determining operational validity. Operational validity ensures "the model's output behavior has the accuracy required for the model's intended purpose over the domain of its intended applicability" (Sargent, 1991:39). Most validation occurs during operational validity testing because any problems found in the computerized model could be due to a faulty conceptual model, improper programming, or incorrect data. Thus, operational validity, in a sense, encompasses conceptual model validity, computerized model verification, and data validity (Sargent, 1991:39-42).

Balci also divides validation techniques into two categories: subjective and statistical. He lists 13 subjective validation techniques and 18 statistical validation techniques (Balci, 1989:67-68). Animation is not listed as a validation technique. In fact, Balci does not mention animation at all. Banks divides validation techniques into subjective and statistical, too (Banks, 1989:550-551). However, he does not list animation as a validation technique either.

A three-step approach for validation is offered by Carson. He places face validity at the beginning of his validation process. Face validity is a process which involves questioning those knowledgeable about the system modeled to determine whether a model or its operation is consistent with the system modeled. Carson then recommends validation of the model assumptions and model output. In contrast to Balci and Banks, though, Carson emphasizes the need for animation. He concludes with a listing of verification and validation techniques (Carson, 1989:555,557).

Animation. There are many proponents for using animation, not only for model validation, but also for the whole model building process. Those who tout animation the most are those who sell animation software (Brunner and others, 1991:90-94; Kalasky and Davis, 1991:123; Hollocks, 1984:322-328; Standridge, 1986, 121-143). Users of animation are the second most enthusiastic proponents of animation (Aiken and others, 1990:775-783; Johnson and Poorte, 1988:30-36; Carson and Atala, 1990:798-801). Academia, although acknowledging value in animation, generally stresses other methods of validation or does not explicitly mention animation at all. (Law and Kelton, 1991:242; Balci, 1989:62-71; Banks, 1989:549-551).

As an example of the latter, Law and Kelton recognize that animation is increasing the popularity of simulation modeling. However, with regard to model validation, they point out that animation can show that a model is not valid, but it cannot show that a model is valid. Also, animation cannot replace statistical analysis of the model output. Another drawback is that animation usually cannot be watched for the complete run of the simulation. Therefore, an error could occur at a point later in the simulation run when there is no one watching. A final drawback is "only part of a simulation model's logic can actually be seen in an animation; thus, a 'correct' animation is no guarantee of a valid or debugged model" (Law and Kelton, 1991:242).

Animation can improve face validity according to Schuppe. However, developing quality animation can take as long as developing the model, and the animation should be verified. That is, the modeler should substantiate that the animation is not giving a false picture of what the simulation model is doing (Schuppe, 1991:523, 525).

Carson paints a much brighter picture for animation. He states that animation provides a degree of validity that previously could not be reached for simulations involving items travelling through space. With an animator spatial relationships are easier to verify. He recognizes there is a possibility judgments will be made based on animation before the validation process is complete. However, Carson asserts the probability of this happening is not high because those who use simulations "will want to see the output data of numerous long runs before making any final judgments" (Carson, 1989:555).

According to Johnson and Poorte, animation can be useful for debugging and verification, validation, analysis, and communication and presentation. They offer a hierarchical approach for using animation in all of these areas. With regard to validation, they believe "animation can provide a vital link between the modeler and expert" (Johnson and Poorte, 1988:31). Usually the system expert involved in conceptual model validation is unfamiliar with simulation code; thus, animation improves the communication between the modeler and the expert, and conceptual model validity is enhanced. Nevertheless, they acknowledge "animation is a supplementary tool and should not be viewed as a replacement for standard analysis techniques" (Johnson and Poorte, 1988:31,36).

Cyr states that animations of simulated activities offers several benefits:

- Animation can help explain simulations (in particular, Monte Carlo simulations) to upper-level managers, who are not experts in mathematics and statistics.
- Animation can help the manager and others understand the system being modeled.
- Animation shows actual or potential system problems that might be hard to understand or might be missed by looking at lists of numbers.
- Animation opens up the opportunity to interact with the simulation.
- Animation lowers the development costs of models by uncovering problems that would be hard to see without animation.

He adds that the value of a simulation models increases when animation is added, and the additional cost is outweighed by the advantages of animation (Cyr, 1992:1000, 1002).

As those involved with the making and marketing of animation software, Kalasky and Davis boldly declare,

> In recent years animation has become a requirement of the simulation process. One of the reasons for this requirement is that numeric summary statistics do not necessarily convey information about the dynamic interactions of components of a system. Although summary statistics are a crucial part of evaluating the performance of a simulated system, it is only through animation that the analyst can easily identify the system status under which, for example, bottlenecks occur. (Kalasky and Davis, 1991:122)

Concerning validation, they agree with Johnson and Poorte that animation improves communication between the system experts and the modelers. In addition, they state that simplifying assumptions can be seen easier with animation. (See also Standridge, 1986:121.)

To conclude their listing of the uses of animation they give the standard qualifier: "animation cannot replace standard statistical analysis techniques" (Kalasky and Davis, 1991:122-123).

Analytic Hierarchy Process. The AHP is a decision making aid. It allows a subject to subjectively compare alternatives. A subject compares every alternative with every other possible alternative. Thus, the AHP is a pairwise comparison. Ratings for each alternative can be calculated from the pairwise comparison data. This is in contrast to absolute estimation methods (such as an ordinal ranking of alternatives). For example, the pairwise comparison of the animations allowed a subject to rate a specific animation's ability to communicate the model's operation against the ability of every other animation. This permitted the relative merits of the animations to be quantified. Since there were seven animations, the subjects performed 21 pairwise comparisons each. (There were not 28 comparisons because an alternative is not compared with itself.)

The AHP was developed by Thomas Saaty (Saaty, 1980). The advantages of the AHP, as summarized by Vidulich and Tsang (1987), follow.

> In comparing the AHP approach to the absolute estimation methods, Saaty (1980) suggested that the AHP has the following advantages: (1) While the number of decisions are more numerous ([$n(n-1)/2$] as opposed to [n] for n conditions to evaluate), they are simpler because the subject can focus on the relationship between only two conditions. (2) The comparison of each condition to every other provides a great deal of redundant information to improve reliability. (3) It is possible to calculate the "consistency" of each subject's pairwise matrix and thereby test the subject's ability to make the discriminations necessary. (Vidulich and Tsang, 1987:1058)

Given n alternatives, the pairwise matrix (or judgment matrix) is an $n \times n$ matrix with each row and column headed by the choices. The numerical rating from each pairwise comparison is placed in the matrix, and the principal eigenvector for the matrix is calculated, which gives the rating for each alternative (Vidulich and Tsang, 1987:1058). There are three steps to using judgment matrices: collect the judgment data, construct the judgment matrices, and calculate the ratings. Vidulich provides a good overview of this process (Vidulich, 1989:1407).

Williams and Crawford offer another method of calculating ratings. They suggest using a geometric means approach instead of an eigenvector approach (Williams and Crawford, 1985:1). They state,

> The geometric mean vector $\mathbf{v} = \mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_n}$, given by
>
> $$v_i = \prod_{j=1}^{n} a_{ij}^{1/n}$$
>
> which satisfies the continuity and consistency criteria Saaty uses to defend the dominant eigenvector, has several other desirable traits: In certain circumstances, it is statistically optimal and gives rise to an estimate of scales and a measure of consistency that have known statistical distributions. In empirical studies reported here it seems to do as well as, or better than, the eigenvector in preserving rank order. In addition, it is supported by a literature describing methods of handling a wealth of variations of the problem, including missing data and multiple judges. (Williams and Crawford, 1985:vi)

Another issue is whether judgment matrices are consistent. Vidulich defines a consistent matrix as one where there are "transitive trends among related judgments" (Vidulich, 1989:1407). For instance if alternative A is preferred twice as much as alternative B, and alternative B is preferred three times as much as alternative C, then alternative A should be preferred six times as much as alternative C. If these types of relationships hold within the matrix, the matrix is considered consistent. Inconsistency increases as these relationships are violated (Vidulich, 1989:1407). Budescu and others give S^2, the measure of consistency developed by Williams and Crawford.

$$S^2 = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left[\ln\left(r_{ij}\right) - \ln\left(GM_i/GM_j\right)\right]^2}{(n-1)(n-2)} \tag{1}$$

where r_{ij} is the pairwise comparison ratio for the ith and jth alternatives and GM_i and GM_j are the geometric means of rows i and j respectively (Budescu and others, 1986:71).

Budescu and others developed a consistency criterion for the geometric means approach during their comparison of the two methods (Budescu and others, 1986:69-78). In his own comparison the eigenvector approach and geometric mean approach, Vidulich discusses the rule developed by Budescu and others and reviews the comparison of the two approaches performed by Williams and Crawford and Budescu and others. He states

> Williams and Crawford (1980) did a Monte Carlo comparison of the eigenvector and geometric mean approaches to judgment matrix analysis. They used matrices that were selected from known distributions with varying degrees of error perturbations introduced. Williams and Crawford found that the two approaches gave similar results when perturbations away from consistency were minimal. But for large perturbations, the geometric means approach produced results that deviated less from the known distribution. The relative advantage of the geometric means procedure increased with both matrix size and error variability. (Vidulich,1989:1408)

Matrices of randomly generated data were used by Budescu and others to compare the two means of calculating ratings. Budescu and others showed that S^2 remained stable as matrix size increased. Vidulich suggested at the end of his study that the geometric mean approach can be used instead of the eigenvector approach. Also, Vidulich recommended that when using S^2 as a consistency criterion, the matrix size should range from 6×6 to 10×10 (Vidulich, 1989:1410).

Conclusion

Many techniques are offered for model validation. These techniques are generally divided into subjective and objective (statistical) techniques. Animation, when listed for use in model validation, is considered to be a subjective technique. Even though animation is being used in simulation modeling more and more, the accepted range of animation's applicability to simulation model validation varies from helpful to almost indispensable. However, animation can never replace statistical validation techniques.

The Analytic Hierarchy Process is a useful means of comparing alternatives. Although there is debate concerning the method of calculating ratings (eigenvector versus geometric means), the geometric means approach is easier to use and has a more stable measure of inconsistency.

III. Methodology

Approach

The experiment consisted of subjects viewing seven different types of animations. The subjects determined how useful the animations were in communicating model operation and which animation type was most beneficial. This was measured in two ways. First, each animation showed a different scenario. The subject attempted to determine if there were any efficiency problems with the system modeled (queue buildups, long idle times, and the like). The time delay in identifying a potential problem and the correctness of problem identification were recorded. Second, after viewing the animations, the subject performed a pairwise comparison of each animation's contribution to face validity. A rating for each animation was calculated from the pairwise comparisons (AHP) which measured the subject's preference for each animation type. Preparation for the experiment included selecting a simulation model, animating the model, creating system problems (different scenarios), constructing the experimental design, and preparing data collection forms. The next step was to perform the experiment and collect data by showing the animations to AFIT faculty and student volunteers. The final step was analyzing the data.

Initialization

Simulation Model. The Loader model is a simple SLAM network simulation that models a loading and hauling operation for 480 minutes (8 hours). The system modeled consists of one bulldozer, four trucks, and two man-machine loaders. The bulldozer stockpiles material for the loaders. Two piles of material must be stocked prior to the initiation of any load operation. In addition to the two loads of material, a loader and an unloaded truck must be available before the loading operation can begin. The time to bulldoze a load is Erlang distributed and is the sum of two exponentials each with a mean of 4 minutes. The loaders are modeled as servers with loading time for server 1 exponentially distributed with a mean of 14 minutes and loading time for server 2 exponentially distributed with a mean of 12 minutes. When a truck has been loaded, it hauls the material to the dumping area, dumps its load, then returns for more material. Hauling time is normally distributed with a mean of 22 minutes and standard deviation of 3 minutes. The time to dump is uniformly distributed between 2 and 8 minutes, and return time is normally distributed with a mean of 18 minutes and standard deviation of 3 minutes. The loader must rest 5 minutes after loading a truck (Pritsker, 1986:237). The SLAM network code and a SLAM network diagram are shown in Appendix C. Figure 1 is a diagram of the simulation.

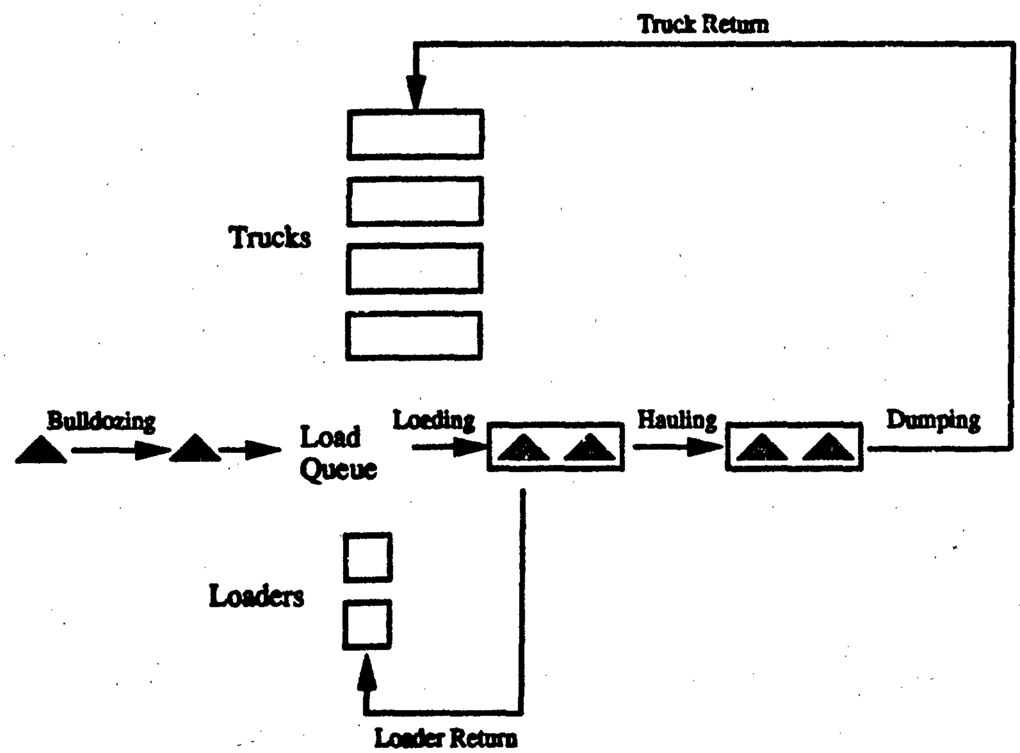


Figure 1. Loader Model Diagram.

The Loader model was changed to ease the process of animating. In the original model there was no way to keep track of the trucks and loaders; however, that information was needed for the animations. Also, the original model allowed two trucks to be loaded at the same time and more than one truck to dump at the same time. Finally, the loader with the longest idle time was selected when a loading operation could begin.

The first problem was solved by assigning a truck number (1, 2, 3, or 4) to attribute 10 of the truck entities and assigning a loader number (5 or 6) to attribute 11 of the loader entities. The loaders and trucks were placed into the appropriate queues at the beginning of the simulation using ENTRY statements. Each load entity had the following attributes assigned:

- Attribute 1 - Not used (originally was to be used as a load counter)
- Attribute 2 - Next load arrival time
- Attribute 3 - Not used
- Attribute 4 - Loader 1 load time
- Attribute 5 - Loader 2 load time
- Attribute 6 - Loader rest time (always 5)
- Attribute 7 - Truck hauling time
- Attribute 8 - Truck dumping time
- Attribute 9 - Truck return time

Since two loads were accumulated at the Accumulate node, only the attributes of the second load entity arriving were saved. When two loads, a truck, and a loader were available the Select node summed the attributes of all three creating a new entity with the above attributes plus the truck number and loader number in attributes 10 and 11 respectively. There was one resource representing the loading area so the new entity had to wait until that resource (LOADSPOT) was available. This solved the problem of two trucks being loaded at once. The appropriate loader server was selected based on Attribute 11. The resource LOADSPOT was then freed, and the entity was cloned with the loader being delayed for 5 minutes. All of the attributes of the loader, except for Attribute 11 (loader number), were zeroed out before the loader was routed back to the loader queue. The second entity encountered the hauling delay, waited for the resource DUMPSPOT (of which there was one), delayed for dumping, freed DUMPSPOT, and then had its return delay. Before this entity was routed back to the truck queue, all of its attributes except Attribute 10 (truck number) were zeroed out. Therefore, assigning attributes allowed tracking of the trucks and loaders, and the resources limited loading and dumping to one truck at a time. The selection of loader based on longest idle time changed to alternating loaders. Reasons why these changes were needed will be given when the animations are discussed because the problems were not encountered until the animations were begun. The diagram in Figure 1 was still correct even after the modifications. These modifications significantly increased the length of the network code, plus Fortran subroutines were needed to create the trace file for Proof. The modified network code is shown in Appendix C.

Animations. Seven different animations were created to examine movement, color, and detail of icons:

- M - Movement. Simple icons that move but do not change level of detail or color.
- I - Icon. Icons exhibit differing levels of detail but do not move or change color.
- C - Color. Simple icons that change color but do not move or change level of detail.
- MI - Movement and Icon. Icons move and change level of detail but do not change color.
- MC - Movement and Color. Simple icons that move and change color but do not change level of detail.
- CI - Color and Icon. Icons change color and level of detail but do not move.
- MCI - Movement, Color, and Icon. Icons move and change color and level of detail.

Two files are required to run a Proof animation: a layout file (.LAY) and a trace file (.ATF). With regard to the layouts, the only background object created was an object representing a pile of material at the dumping area. This object only appeared in MI and MCI. Any object on the screen that could move or change status in some way required a unique number. That is why it was necessary to keep track of the trucks and loaders in the simulation model. Object classes were created that represented loaded and unloaded trucks and loaders. When a particular object was needed, a class was assigned by the trace file to the object's number. The object that appeared on the screen was based on the class assigned to the object's number. An object's color, speed, or travel time could also be changed by assigning a different color or speed to the object's number. For instance, "SET 10 COLOR RED" where 10 is the object number, or "SET 10 TRAVEL 9.2" where 9.2 is 9.2 minutes are examples of trace file commands that change object status. Figure 2 shows the different objects classes used.

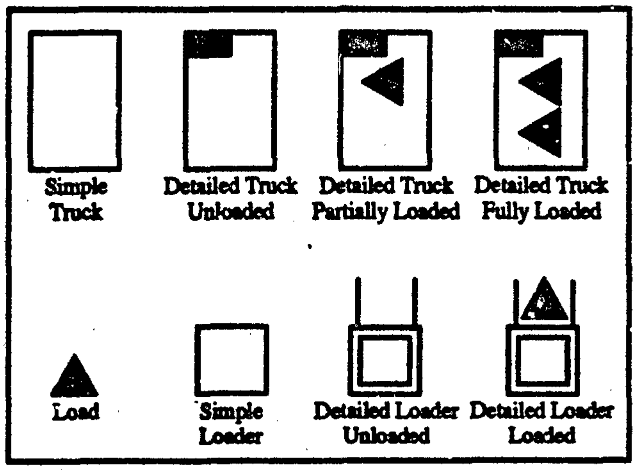


Figure 2. Animation Object Classes.

For the animations with movement, paths were created for the objects to follow. Allowing two trucks to load or dump at the same time greatly complicated the path structure and the logic for path usage. That is why the trucks were limited to loading and dumping one at a time. Also, the purpose of this research was to examine animations, not modeling assumptions. A snapshot of an animation with movement is in Figure 3. The icons shown are the ones for MI and MCI. No bulldozer icon was used. The bulldozer was represented by the loads arriving. The loading times were divided by six and assigned to a portion of the loading process. For example, if the loading time was 12 minutes, the following would occur:

- The loader would leave its idle position and travel for two minutes to pick up a load.
- The loader class would change to loaded, and the loader would travel for two minutes to the truck.
- The loader would wait at the truck for two minutes to represent the transfer of the load to the truck.

- The loader class would change to unloaded and then travel for two minutes to pick up the second load, and the truck would change class to show one load.
- The loader class would change to loaded, and the loader would travel for two minutes to the truck.
- The loader would wait at the truck for two minutes to represent the transfer of the second load to the truck.
- The loader class would change to unloaded and then travel for five minutes to its idle position, and the truck would change class to show two loads and begin hauling.

The two loads "fell" out of the truck onto the pile when dumping. M and MC did not show the pile of material, and the icons were the ones labeled "Simple Truck" and "Simple Loader" in Figure 2; however, they did have the same movement. For MC, the icons were white when idle, green when travelling empty, red when travelling loaded, pink when partially loaded (trucks only), and yellow when dumping (trucks only). MCI had the same color scheme with the icons as described above. The icons remained red for M and MI.

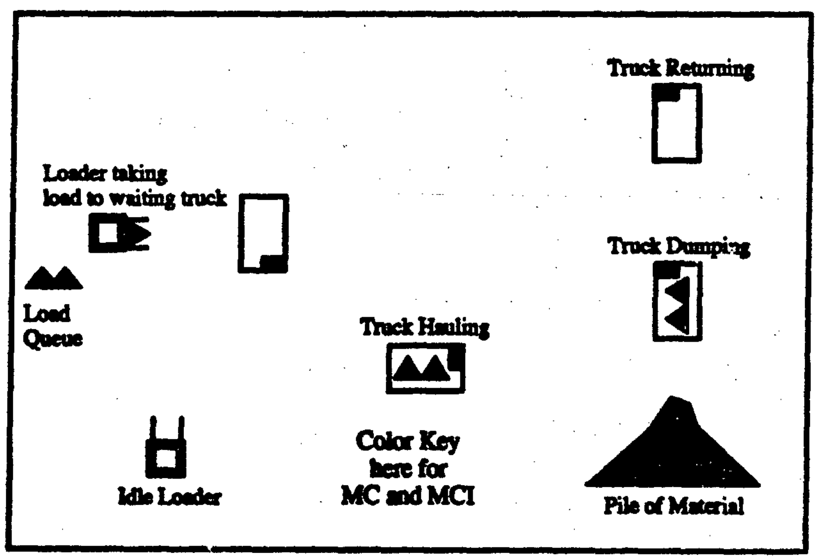


Figure 3. Sample Animation With Movement.

The animations without movement used larger stationary icons. The four trucks were displayed at the top of the screen and the loaders at the bottom of the screen. Figure 4 gives a representation of the animations without movement (C, I, and CI). The figure shows the icons for I and CI. As with M and MC, the icons for C were the ones labeled "Simple Truck" and "Simple Loader" in Figure 2. For C and CI, the color changes were the same as for MC and MCI, and the icon changes for I and CI were the same as for MI and MCI. The icons in I remained red. There was no representation of the load queue in the stationary animations.

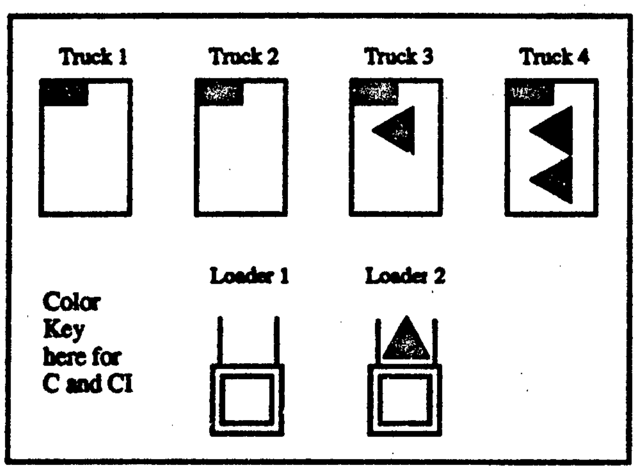


Figure 4. Sample Animation Without Movement.

Scenarios. Problems were incorporated into the experiment to motivate the subjects to concentrate on the animations and to measure the performance of the different animations. The subjects were the "system experts" (owner, operator, or manager of the loading operation); thus, they were looking for efficiency problems with their system. Table 1 gives the scenario abbreviation, the simulation model modification, and the associated system problems for each scenario.

Experimental Design. The order in which the animations were viewed was randomized as well as the scenario associated with each animation. Therefore, there were 49 different animation and scenario combinations. Table 2 shows the experimental design

Table 1. Scenario Descriptions.

Scenario	Model Modifications	System Problems
LDER	Load interarrival time, loading time, hauling time, dumping time, and return time unchanged. (Original Model with Modifications)	All resources are used adequately with a slight buildup of loads.
SLTK	Hauling time, dumping time, and return time doubled for the third truck. (Slow Truck)	One truck is much slower than the others, and there is a buildup of loads.
SLLD	Loading time doubled for the first loader. (Slow Loader)	One loader is too slow, which creates a buildup of loads and idle time for the trucks.
FT	Hauling time, dumping time, and return time cut in half for all trucks. (Fast Trucks)	There are too many trucks for the number of loaders and bulldozers or not enough loaders and bulldozers for the number of trucks.
FL	Loading times halved for both loaders. (Fast Loaders)	There are too many loaders for the number of trucks and bulldozers or not enough trucks and bulldozers for the number of loaders.
SL	Load interarrival time multiplied by 2. (Slow Loads)	There are not enough bulldozers, which creates idle time for the trucks and loaders.
ST	Hauling time, dumping time, and return time doubled for all trucks. (Slow Trucks)	There are too few trucks, which creates a buildup of loads and loader idle time.

constructed for 50 subjects. All 49 animation and scenario combinations had been selected by run 32. A run consisted of the seven animation and scenario combinations to be viewed by a subject.

Data Collection Forms and Pictures.

Forms. Several forms were needed for this experiment, all of which are given in Appendix A. The first form created was the Scenario/Animation Viewing form. At the top were places for the subject's name, the date, and the run number. Also, there was a selection for student, faculty, or other. Following this there were seven groups of lines used for listing the scenario, animation, problem identification time, and problem observed. At the bottom of the form was a place for recording the subject's selection of "best" and "worst" animation type. This form was completed by the experimenter while a subject viewed the animations.

The second form was the pairwise comparison form. This form consisted of three pages, the first of which included instructions for completing the form and a listing of the animation types. The second page gave some pairwise comparison examples, and the last page was the form itself. The same information was at the top of this last page as was at the top of the Scenario/Animation Viewing form. The pairwise form contained 21 lines. This allowed for the comparison of each animation type with every other animation type. No animation type was compared with itself. Each line had 17 blanks with the ninth (or middle) blank marked off in a column by itself. An animation type was listed on each side of the line. If the subject considered the animation types to be equal in contributing to face validity (that is, both communicated equally well), he or she would mark the middle column. If one animation type was preferred over the other, then the subject would mark in a space on that animation type's side. The more the subject preferred the animation type, the further out to the side he or she would mark.

The pairwise comparison form provided the data for the pairwise matrix (or judgment matrix), which was the third form. The pairwise matrix is the one described in Chapter 2. The top of the form included the subject's name, the date, and the run number. The

Table 2. Experimental Design.

Run	Scenario/Animation						
1	LDER/MC	SLTK/I	SLLD/M	FT/MI	FL/CI	SL/MCI	ST/C
2	LDER/I	ST/M	SLLD/MCI	FL/C	SLTK/CI	FT/MI	SL/MC
3	LDER/I	FL/C	SL/MI	SLTK/MCI	ST/CI	FT/MC	SLLD/M
4	SLTK/I	ST/C	SLLD/M	SL/MI	FT/CI	FL/MC	LDER/MCI
5	ST/MC	SLTK/C	SLLD/CI	LDER/MI	FL/M	SL/I	FT/MCI
6	FT/M	SLLD/C	SLTK/MCI	LDER/I	FL/CI	ST/MC	SL/MI
7	FT/MCI	FL/C	SL/M	ST/MC	SLTK/I	LDER/CI	SLLD/MI
8	ST/CI	SLTK/I	FT/MC	SL/MCI	FL/C	LDER/M	SLLD/MI
9	SL/MI	ST/I	SLTK/CI	FT/MCI	SLLD/C	FL/MC	LDER/M
10	ST/M	SLLD/I	LDER/MC	SLTK/MCI	FT/MI	SL/C	FL/CI
11	SLTK/MC	SLLD/I	LDER/M	ST/MI	FT/C	SL/CI	FL/MCI
12	ST/MC	FT/CI	SLLD/C	SLTK/M	FL/I	SL/MI	LDER/MCI
13	SL/M	SLLD/MC	SLTK/MCI	LDER/I	ST/CI	FT/MI	FL/C
14	FL/M	ST/C	FT/MCI	SL/MC	SLLD/MI	SLTK/CI	LDER/I
15	FL/I	SL/M	ST/C	SLTK/MC	FT/MCI	SLLD/MI	LDER/CI
16	ST/MC	FT/C	SL/M	SLTK/CI	LDER/MI	FL/MCI	SLLD/I
17	FT/MI	LDER/M	ST/C	SLTK/I	FL/CI	SLLD/MCI	SL/MC
18	ST/I	LDER/M	SL/C	FL/MI	FT/CI	SLTK/MC	SLLD/MCI
19	SLTK/MCI	SL/M	LDER/I	FL/MI	SLLD/CI	ST/C	FT/MC
20	LDER/CI	SL/C	FT/M	FL/MI	SLTK/MC	ST/MCI	SLLD/I
21	ST/MC	SLLD/C	LDER/CI	SLTK/M	SL/MCI	FT/MI	FL/I
22	SL/M	LDER/MC	FL/C	ST/MCI	SLTK/MI	FT/CI	SLLD/I
23	SLTK/CI	SLLD/MCI	SL/I	FL/C	ST/MC	LDER/MI	FT/M
24	FT/M	SL/C	ST/MCI	LDER/MI	SLLD/MC	FL/I	SLTK/CI
25	FL/CI	SLTK/C	SLLD/M	FT/MC	ST/MCI	SL/I	LDER/MI
26	SL/M	SLLD/MC	LDER/I	FT/CI	ST/C	FL/MI	SLTK/MCI
27	SL/C	LDER/M	FL/I	SLTK/MCI	FT/CI	ST/MI	SLLD/MC
28	SL/CI	SLTK/MCI	ST/C	SLLD/MI	LDER/M	FL/I	FT/MC
29	SLLD/I	SLTK/C	FL/CI	FT/M	SL/MI	LDER/MC	ST/MCI
30	SLLD/MI	LDER/MC	FT/M	FL/MCI	SL/C	ST/CI	SLTK/I
31	SLLD/MCI	FL/MC	SL/M	LDER/C	SLTK/CI	ST/MI	FT/I
32	SLLD/MC	ST/MI	FL/I	FT/MCI	SLTK/C	SL/CI	LDER/M
33	FT/M	FL/CI	SL/MI	ST/MC	LDER/I	SLTK/MCI	SLLD/C
34	LDER/I	SLTK/CI	SL/C	FT/MCI	ST/M	FL/MI	SLLD/MC
35	SLLD/MI	FT/MC	SL/CI	LDER/C	ST/MCI	FL/M	SLTK/I
36	LDER/MC	SLTK/C	FL/M	SL/MI	ST/I	SLLD/CI	FT/MCI
37	FL/CI	ST/C	SLLD/MI	LDER/MCI	FT/M	SL/I	SLTK/MC
38	SLLD/M	LDER/I	FL/MC	SLTK/MCI	ST/CI	FT/C	SL/MI
39	ST/M	SLTK/C	FT/MC	LDER/I	SL/CI	SLLD/MCI	FL/MI
40	SL/M	FT/MCI	SLLD/I	SLTK/MC	LDER/MI	ST/C	FL/CI
41	SLTK/CI	SL/C	LDER/MCI	ST/MC	SLLD/M	FT/MI	FL/I
42	LDER/I	FL/CI	SLTK/MC	SLLD/M	FT/MI	ST/C	SL/MCI
43	SLTK/M	LDER/I	ST/MC	SLLD/MI	FL/C	SL/MCI	FT/CI
44	SL/M	ST/C	FL/MC	LDER/MI	SLTK/MCI	SLLD/I	FT/CI
45	ST/C	SLTK/MCI	LDER/CI	SLLD/I	SL/MC	FT/MI	FL/M
46	LDER/M	SL/C	SLLD/MC	FT/MCI	FL/MI	SLTK/I	ST/CI
47	LDER/CI	ST/C	SLLD/MC	FL/M	SLTK/MI	FT/I	SL/MCI
48	LDER/M	ST/MI	FL/MCI	SL/I	SLLD/MC	FT/C	SLTK/CI
49	SL/MC	LDER/CI	ST/MI	SLLD/C	FT/M	SLTK/I	FL/MCI
50	SL/M	ST/C	LDER/I	FL/MI	SLTK/CI	FT/MC	SLLD/MCI

matrix had the rows and columns headed by the animation types. Ones were placed on the diagonal since it was assumed that an animation type was equal with itself. The lower portion was shaded because the *ji* element is simply the reciprocal of the *ij* element. Places were provided to the side of the matrix for the calculated ratings. The pairwise matrix only is shown in Figure 5. Finally, a description of the model was written for the subject to read. This description included the diagram in Figure 1, a discussion of face validity and the subject's role as a "system expert", and a description of the animation types.

	M	C	I	MI	MC	CI	MCI
M	1						
C		1					
I			1				
MI				1			
MC					1		
CI						1	
MCI							1

Figure 5. Pairwise Matrix.

Pictures. Pictures of each animation type were used at the beginning to explain each animation type, and the subject referred to the pictures when completing the pairwise comparisons of the animations. The pictures were 8×10s and were taken at the same point in simulated time using the LDER scenario.

The Experiment

The experiment was conducted in an isolated room, and a "Do Not Enter" sign was placed on the outside of the door to preclude interruptions. Figure 6 shows the physical setup. The computer was against the wall with the pictures of the animation types attached to the wall above the computer. Control of the computer was maintained by the researcher, who was seated next to the subject. This allowed the researcher to start the animations and the stopwatch at the same time. The subject was verbally given the purpose of the experiment and then asked to read the description of the model. While the subject was reading, the information requested at the top of each form was filled in by the researcher, and a run number was assigned. Once the description was read and questions were answered (if any), the animations were described using the pictures. Each animation was viewed for one minute, which equated to 350 simulated minutes. The stopwatch was started when each animation began. This allowed the time to be recorded if the subject identified a problem. If no problem was observed, 60 seconds was entered for the problem identification time and "no problem observed" was entered for the problem observed portion of the form. After viewing all the animations, the subject performed the pairwise comparisons. The subjective selection of the animation types that communicated the "best" and the "worst" concluded the experiment. The total time required was 25 to 30 minutes. The pairwise matrix was filled in, and the ratings were calculated after the subject left.

Analysis Methods

The analysis was divided into three sections based on the type of data. Summary statistics such as numbers and percentages were calculated for the subjective evaluation data (selection of "best" and "worst" animations). The AHP ratings were calculated using the SWORD computer program provided by Vidulich (SWORD, 1989). This program also calculated the S^2 value and the critical S^2 value. Simple statistics were calculated for the AHP ratings; plus, Analysis of Variance, Factor Analysis, and Cluster Analysis was performed. The third set of data was the problem data. Simple statistics and summary statistics were calculated for this data also. In addition, the data was analyzed using Anal-

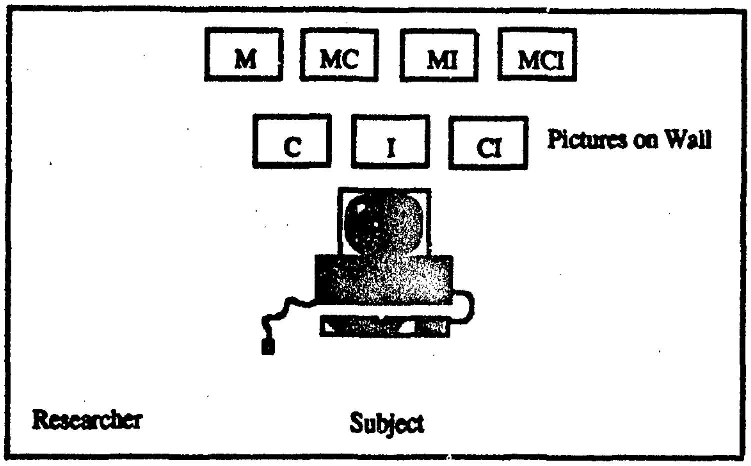


Figure 6. Room Setup.

ysis of Variance and Cluster Analysis from the animation perspective. Analysis of Variance was used for analyzing the problem data from the scenario perspective. The animation perspective looked at the performance of the subjects when viewing the animations with regard to time of problem identification and accuracy of problem identification by isolating the animation type. The scenario perspective examined the animation types by isolating the scenario.

IV. Results and Discussion

A total of 47 individuals volunteered to view the animations. Of the 47, 41 were AFIT students, five were AFIT faculty, and one was neither. The results from the subjective evaluation, including the "best" and "worst" selections and the pairwise comparison (AHP ratings), will be presented first. The results of the objective evaluation (problem identification accuracy and times) will follow, and a comparison of the subjective and objective results will conclude the chapter.

Subjective Results

"Best" and "Worst". Only three animation types were chosen as communicating the best: MCI, MI, and M. Two animation types were identified as worst: C and I. Table 3 gives the results for each animation type.

Table 3. Number of "best" and "worst" animation types.

Rating	C	I	CI	MC	M	MI	MCI
Best	0	0	0	0	1	9	37
Worst	36	11	0	0	0	0	0

The most preferred animation type was MCI, which was selected as the best 37 times (79%). Chosen 36 times (77%) as the animation type preferred the least was C.

AHP Ratings. All of the judgment matrices' S^2 values except one passed the consistency criterion. However, the judgment matrix that did not pass was used. This is because the S^2 value was only slightly above the critical value. Discussions with Vidulich revealed that slightly inconsistent matrices did not affect the analysis enough to warrant not using the data (Vidulich, 1992). Table 4 gives the mean and standard deviation of the ratings for each animation type while Figure 7 is a scatter plot of the ratings. There is a clear distinction between animation types without movement and animation types with movement. The subjects demonstrated the highest preference for MCI. However, most stated that movement was the key to understanding the model.

Table 4. **Animation Rating Means and Standard Deviations.**

Statistic	C	I	CI	M	MC	MI	MCI
Mean	0.0287	0.0353	0.0561	0.1221	0.1693	0.2340	0.3544
St Dev	0.0195	0.0130	0.0243	0.0576	0.0478	0.0693	0.0658

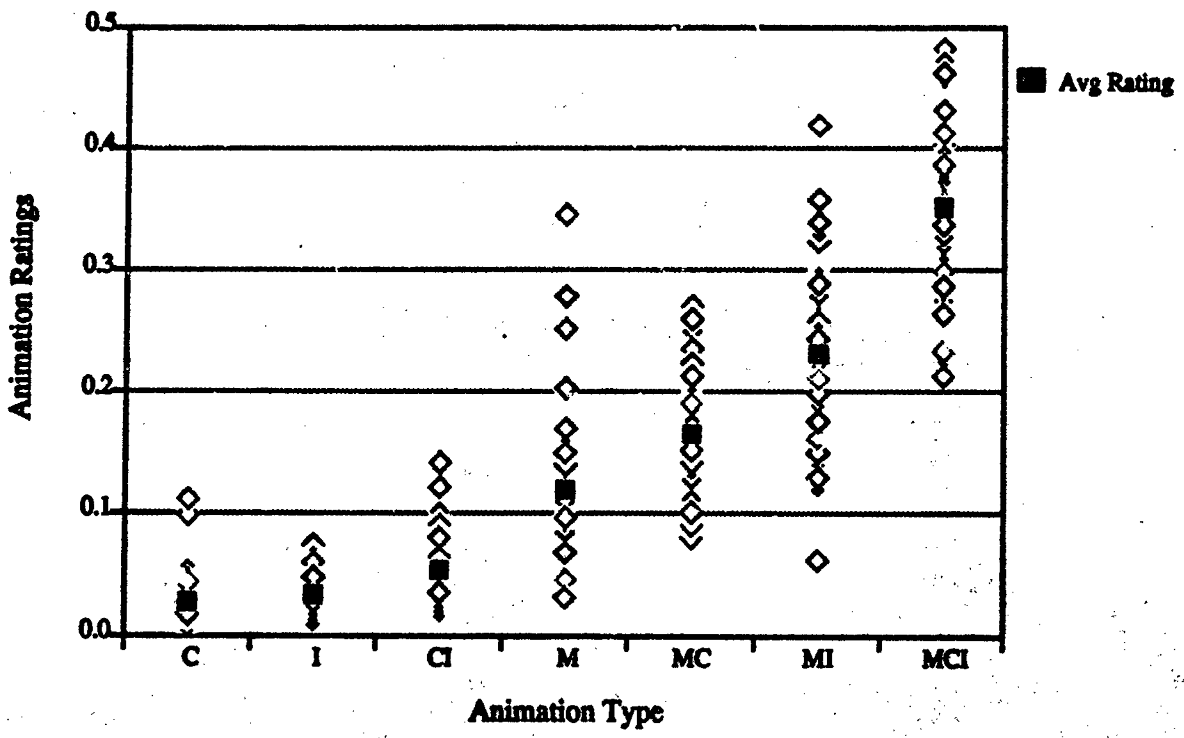


Figure 7. Scatter Plot of Animation Ratings.

AHP ANOVA. This Analysis of Variance assumed a model of the form

$$Y_{ij} = \mu + T_i + B_j + \varepsilon_{ij} \tag{2}$$

where:

- μ is the mean of the responses
- T_i are constants for the treatment effects
- B_j are constants for the block effects
- ε_{ij} are independent $N(0, \sigma^2)$
- $i = 1, \ldots, 7; j = 1, \ldots, 47$

Thus, the ratings Y_{ij} are assumed to be independent and normally distributed, with mean and constant variance

$$\begin{aligned} E\{Y_{ij}\} &= \mu + T_i + B_j \\ \sigma^2\{Y_{ij}\} &= \sigma^2 \end{aligned}$$

Each animation type was a treatment ($i = 1, \ldots, 7$), and each subject was a block ($j = 1, \ldots, 47$). So this model determined if there were differences among the animation types and differences between subjects. Table 5 is the ANOVA table.

Table 5. Analysis of Variance Table for Animation Ratings.

Source	DF	SS	MS	F	P
Treatment	6	4.05716	0.67619	255.61	0.0000
Block	46	3.429E-06	7.453E-08	0.00	1.0000
Error	276	0.73014	0.00264		
Total	328	4.78731			
Grand Average	1	6.71428			

The ANOVA table shows that there is a difference between animation types. The interesting result, though, is that there is no block effect. That is, there was no significant difference between subjects. A Tukey test for additivity was performed to test for interaction effects between animation type and subject. Since the null hypothesis of the test is no interaction effects (treatment and block are additive), the F value of 0.04 and the P value of 0.8370 indicated there was no animation type and subject interaction effects. Table 6 gives the complete results of the test.

Table 6. Tukey's 1 Degree of Freedom Test For Additivity.

Source	DF	SS	F	P
Nonadditivity	1	1.126E-04	0.04	0.8370
Remainder	275	0.73003		

Figure 8 shows a normal probability plot. The residuals were plotted against the expected residuals under normality to check the normality assumption. The almost straight line indicates that the normality assumption is valid. Another requirement for ANOVA is equal variances for the animation types (treatments). The residuals are plotted against the animation types in Figure 9. The plot seems to show different variances. However, there is a good scattering of the residuals in Figure 10, which shows the residuals versus the subjects (blocks).

Since the plot of the residuals versus treatment for (2) seemed to indicate different variances, a Bartlett's test of equal variances was performed after dropping the block variable. The test indicated that the constant variance assumption for the ANOVAs was not valid. The test yielded a high χ^2 value (180.31) and a P value of 0.0, so the null hypothesis of equal variances for each animation type was rejected. The results of the test are shown in Table 7. However, because the sample sizes are the same for each animation type, the model still can be used. Neter and others assert that the effects of unequal variances are minimized when the sample sizes are the same. They state, "When the error variances are unequal, the F test for the equality of means with the fixed ANOVA model is only slightly affected if all factor level sample sizes are equal or do not differ greatly" (Neter and others, 1990:624).

The Tukey method of multiple comparisons was used to determine if the differences between the mean ratings of each animation type were statistically significant. The means

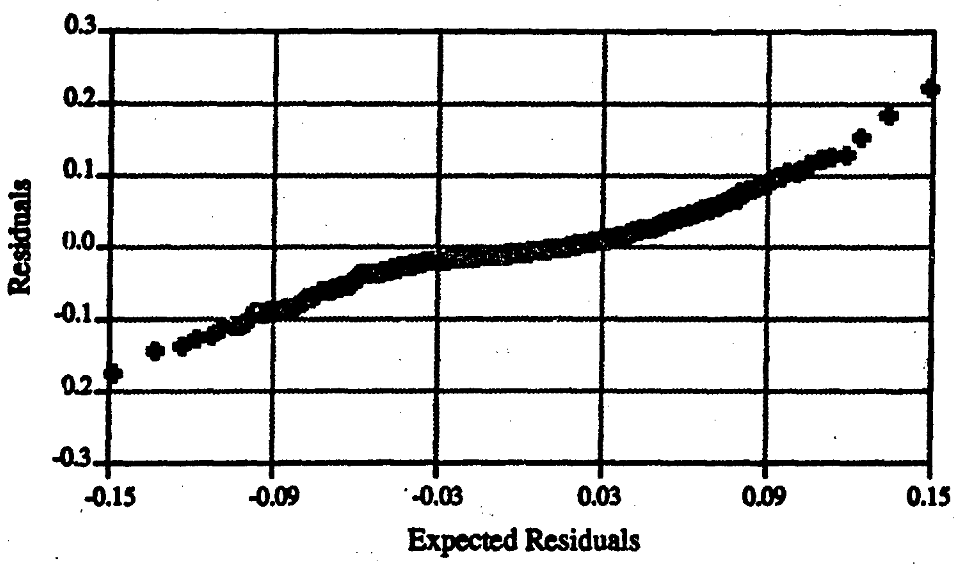


Figure 8. Normal Probability Plot of Residuals vs Expected Residuals.

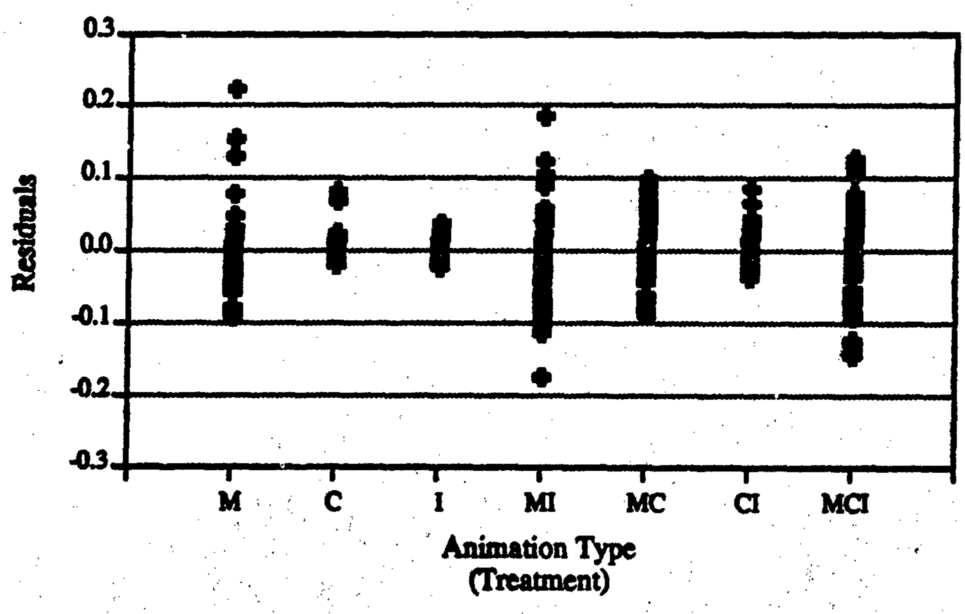


Figure 9. Scatter Plot of Residuals vs Animation Type (Treatment).

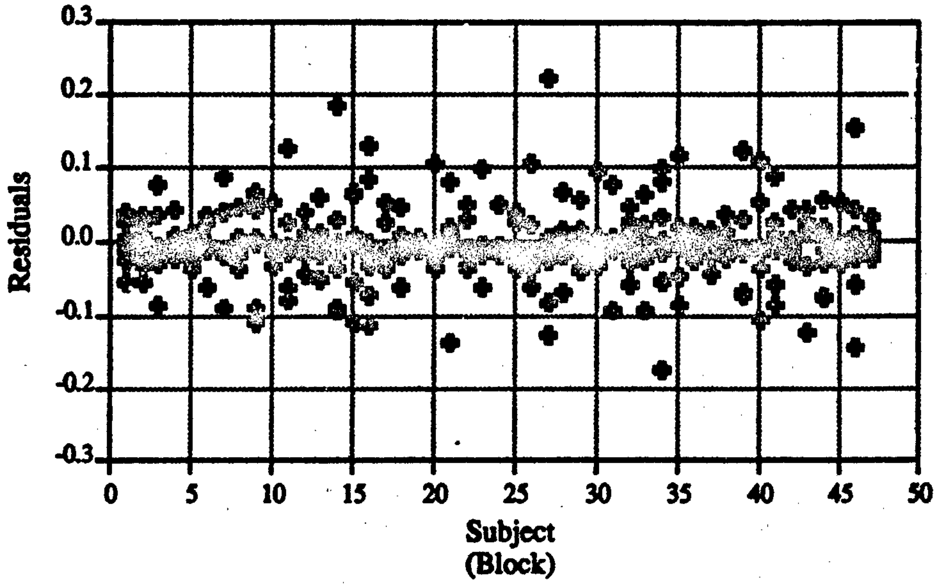


Figure 10. Scatter Plot of Residuals vs Subject (Block).

Table 7. Bartlett's Test of Equal Variances for Animation Type.

Chi-Square	DF	P
180.31	6	0.0000

were tested at a family α level of 0.1. The differences between the mean ratings of the animations without movement (C, I, and CI) were determined not to be statistically significant. That is, at the 90% confidence level the means were considered to be the same. Each of the mean ratings for the animation types with movement (M, MC, MI, and MCI) were statistically different from each other and the group of animation types without movement. Thus, the test determined that there were five groups: MCI, MI, MC, M, and (C, I, CI).

Since there was no block effect, and there was no treatment and block interaction effects, block was dropped. So the model became

$$Y_{ij} = \mu + T_i + \varepsilon_{ij} \tag{3}$$

where:

- μ is the mean of the responses
- T_i are constants for the treatment effects
- ε_{ij} are independent $N(0, \sigma^2)$
- $i = 1, \ldots, 7; j = 1, \ldots, 47$

Again, the ratings Y_{ij} are assumed to be independent and normally distributed, with mean and constant variance

$$\begin{aligned} E\{Y_{ij}\} &= \mu + T_i \\ \sigma^2\{Y_{ij}\} &= \sigma^2 \end{aligned}$$

Table 8 is the ANOVA table for (3). This ANOVA also indicated that there is an effect based on animation type. That is, there is a difference in preferences for animation types. So the ANOVA and associated tests show that animations without movement are preferred the least. Since these animations did not communicate as well as the animations with movement, they were not considered to be high contributors to understanding the model. Understanding the model operation aids in assessing face validity. Preference for animations with movement varied based on what other aspects of animation were included (color and icon). Movement was judged as a necessary contributor to face validity, but adding other aspects were judged to improve the performance of movement in communicating the operation of the model.

Table 8. One-Way Analysis of Variance Table for Animation Ratings.

Source	DF	SS	MS	F	P
Between	6	4.05716	0.67619	298.21	0.0000
Within	322	0.73014	0.00226		
Total	328	4.78731			

AHP Factor Analysis. The sample covariance matrix was used for the factor analysis since all the ratings were in the same units (preference). Three factors were retained because 93% of the variance was explained by the first three factors. The eigenvalues, proportion of variance explained, and cumulative proportion of variance explained are given in Table 9. There was a significant drop from the third to the fourth eigenvalue, and only a small increase in proportion of variance explained. The three factors that were kept were varimax rotated and are shown in Table 10.

Table 9. Eigenvalues of the Animation Ratings Covariance Matrix.

Eigenvalue	0.0081	0.0046	0.0021	0.0008	0.0002	0.0001	0.0000
Proportion	0.5088	0.2872	0.1325	0.0501	0.0131	0.0082	0.0000
Cumulative	0.5088	0.7960	0.9285	0.9787	0.9918	1.0000	1.0000

Table 10. Rotated Animation Rating Factors.

Animation	Factor 1	Factor 2	Factor 3
M	-0.07000	**0.96439**	-0.18944
C	**0.61113**	-0.00739	-0.04354
I	-0.03858	0.02393	-0.27963
MI	**-0.82293**	-0.06075	-0.56269
MC	**0.87741**	-0.36355	-0.13631
CI	0.48971	0.08657	0.27445
MCI	-0.06252	-0.54858	**0.82630**

The animation types which load the highest on Factor 1 are C, MI, MC, and CI. The sign of MI is opposite to that of the others, though. The animation types C, CI, and MC include color and are positive in sign. Although MCI contained color and was negative in sign like MI, it did not load high on Factor 1. Movement loads the highest on Factor 2, and MCI and I load the highest on Factor 3; however, -0.28 for I is too small to be considered significant. In addition, MCI loads significantly on Factor 2, but in opposite sign to M, and MI loads significantly on Factor 3, but in opposite sign to MCI. So Factor 1 appears to be measuring preference for color, in particular, preference for movement with color versus preference for movement with icon. Either a subject liked color or not, and this preference was most evident when color was combined with movement. Factor 2 measures preference for movement alone. Preference for MCI dropped the most as preference for M alone increased. The third factor shows preference for MCI. As the rating for MCI increased, the rating for MI decreased. The plots of the factor scores confirm these conclusions. Figure 11 shows the plot of Factor 1 scores versus Factor 2 scores. The outliers for Factor 1 contrast MC and C with MI. For those on the left, MC and C were rated low, and MI was rated high. For the outlier on the left, MC and C were rated high, while MI was rated low. A look at the subjective evaluations also confirms this. The two on the left that rated MC and C very low chose MI as the best animation instead of MCI. M was rated extremely high for the three outliers at the top for Factor 2, and M was rated low for the outlying point at the bottom. The plot of Factor 1 versus Factor 3 scores is shown in Figure 12. The same outliers are seen for Factor 1. Factor 3's outliers at the top rated MCI very high, while the point at the bottom rated MCI quite low. Figure 13 contains the plot of Factor 2 scores versus Factor 3 scores. The same outliers are seen here, and, again, the scores in which M was rated high are significantly different from the group of scores. Therefore, the pairwise comparison measured preference for color with movement versus icon with movement, preference for movement, and preference for movement, color, and icon.

AHP Cluster Analysis. Three different cluster analysis techniques were used: Complete Linkage, Average Linkage, and Ward's Minimum Variance. All three techniques clustered the animations without movement (C, I, and CI) right away. There were differ-

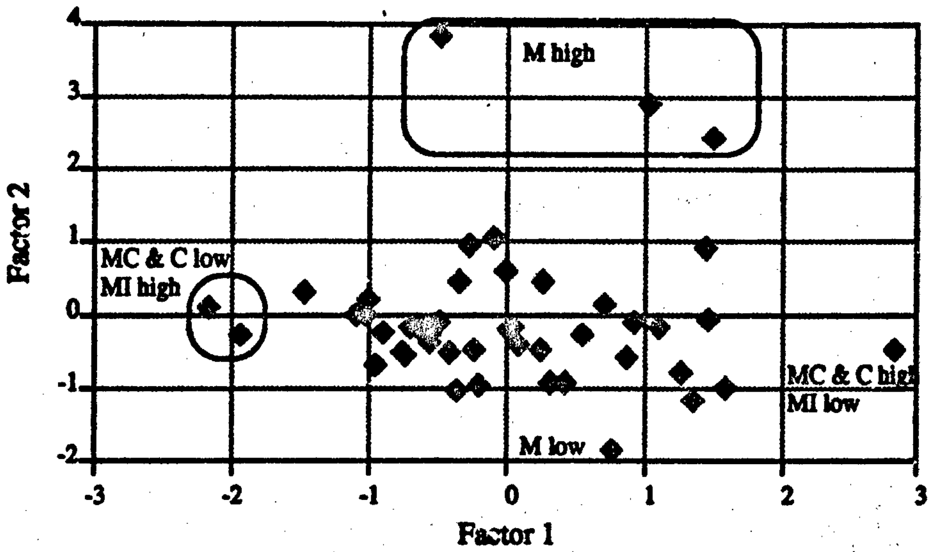


Figure 11. Plot of Factor 1 Scores vs Factor 2 Scores.

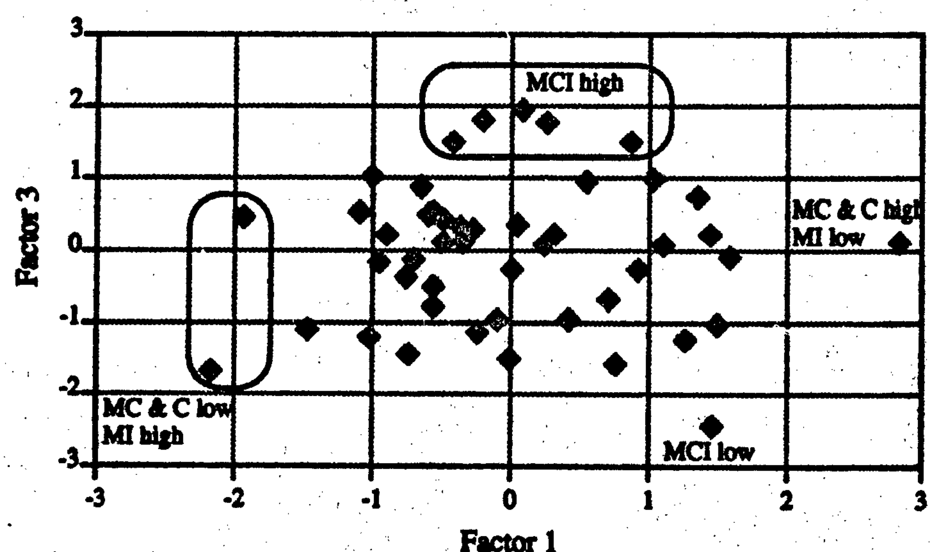


Figure 12. Plot of Factor 1 Scores vs Factor 3 Scores.

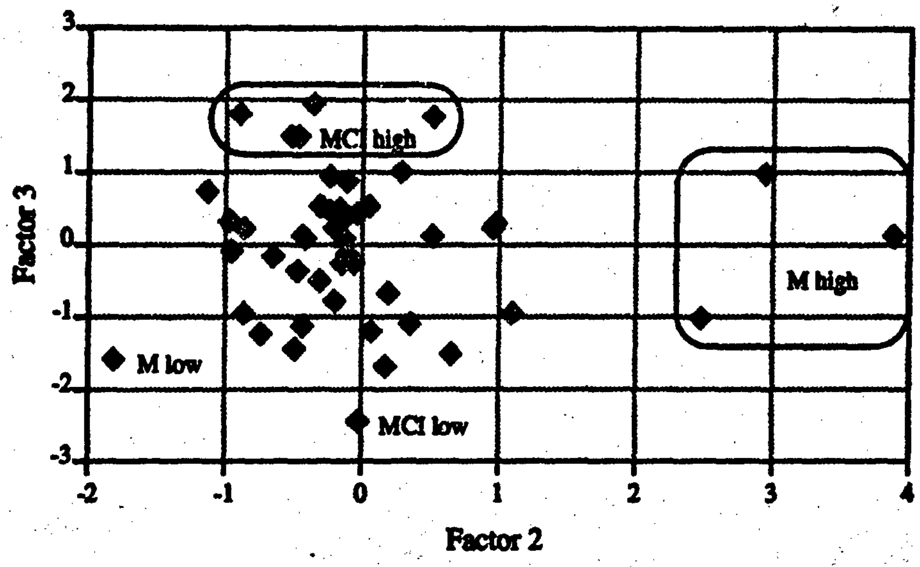


Figure 13. Plot of Factor 2 Scores vs Factor 3 Scores.

ences in how M, MC, MI, and MCI were clustered. The complete linkage technique and Ward's minimum variance technique clustered M and MC, then MI with M and MC. Next, M, MC, and MI were clustered with C, I, and CI. Finally, at a much farther distance, MCI was clustered with all the other animation types. Average linkage was similar except that MI was clustered with MCI instead of M and MC. MI and MCI were then clustered with the remaining animation types at approximately double the distance that the other animation types were clustered. Table 11 shows the clusters and the distance for the complete linkage technique, and Figure 14 is the dendogram for the complete linkage. A dendogram is a graph that shows the groupings and the level at which they were grouped. The distance is the normalized maximum distance between two clusters. Table 12 and Figure 15 show the same for the average linkage technique, and Table 13 and Figure 16 gives the results for Ward's minimum variance technique. The distance measure for the average linkage technique is the normalized root-mean-square (RMS) distance, which is "the root-mean-square distance between pairs of objects in the two clusters joined with one object from each cluster" (SAS Institute Inc., 1989:567). For Ward's technique, the distance measure is the Semipartial R-Squared. This is the decrease in the proportion of variance accounted for after the two groups have been joined and is equal to "the between-cluster sum of squares divided by the corrected total sum of squares" (SAS Institute Inc., 1989:567).

The cluster analysis yielded results similar to Tukey's method of multiple comparisons. There was very little difference in preference for animations without movement. That is, C, I, and CI were not judged to be high contributors to face validity. With regards to movement, the differences in preferences for M, MC, MI, and MCI were large enough for clustering to occur after C, I, and CI were clustered. Also, for complete linkage and Ward's minimum variance, M, MC, and MI were grouped together before they were grouped with C, I, and CI. Thus, movement was judged as significant. As with the factor analysis, average linkage shows that icon with movement was considered to communicate better than color with movement. This is seen in that MI was clustered with MCI instead of M and MC. The primary benefit of the cluster analysis is that it confirmed that all of the animation types without movement were judged to be basically the same. C, I and CI were considered to be poor animations to use in communicating the operation of a model.

Table 11. Complete Linkage Cluster Analysis Results.

Animations Joined	Distance
C, I	0.154549
C, I, CI	0.214065
M, MC	0.604346
M, MC, MI	0.866503
C, I, CI, M, MC, MI	1.364139
C, I, CI, M, MC, MI, MCI	2.062868

Table 12. Average Linkage Cluster Analysis Results.

Animations Joined	Distance
C, I	0.135478
C, I, CI	0.184487
M, MC	0.529770
C, I, CI, M, MC	0.662877
MI, MCI	0.888342
C, I, CI, M, MC, MI, MCI	1.311767

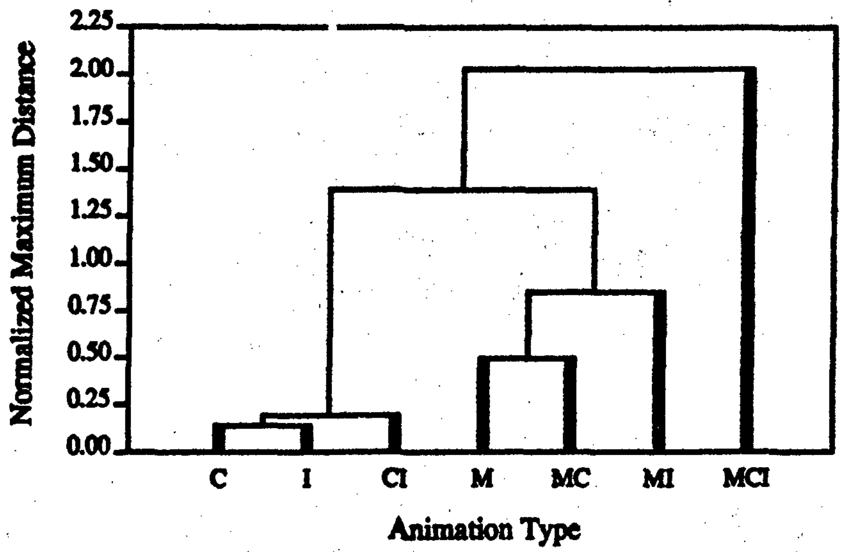


Figure 14. Complete Linkage Dendogram.

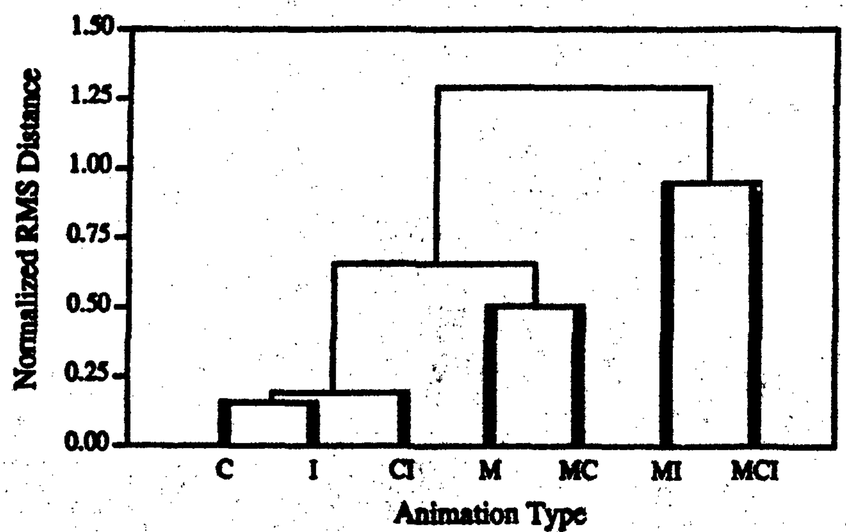


Figure 15. Average Linkage Dendogram.

Table 13. Ward's Minimum Variance Cluster Analysis Results.

Animations Joined	Distance
C, I	0.003059
C, I, CI	0.006544
M, MC	0.046776
M, MC, MI	0.097934
C, I, CI, M, MC, MI	0.284481
C, I, CI, M, MC, MI, MCI	0.561206

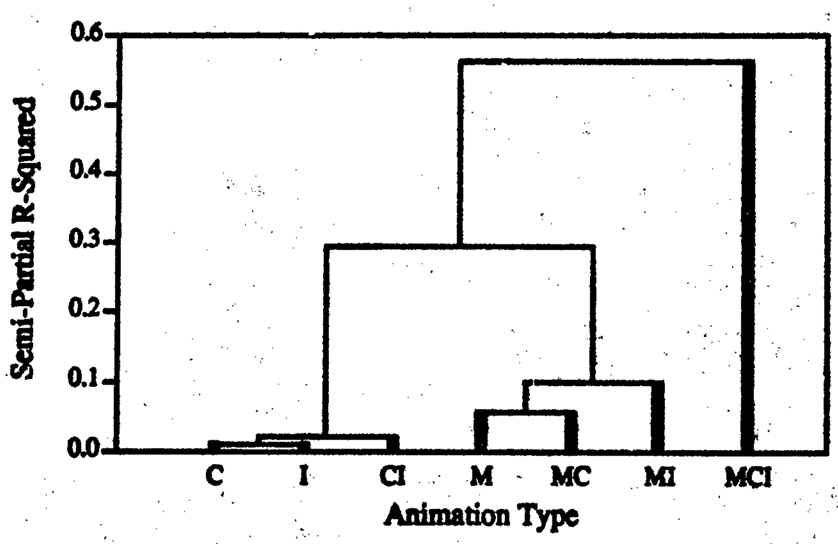


Figure 16. Ward's Minimum Variance Dendogram.

Objective Results

Problem Results From Animation Perspective.

Summary Statistics. Given an animation type, two items were looked at when the problem identification data was analyzed: problem identification time and problem identification accuracy. The AHP ratings measured the subjective preference for an animation type. The problem data measured the subjects' objective performance when viewing the animations. Figure 17 shows the average time (in seconds) that a potential problem was identified, and Table 14 gives the corresponding numbers. The animations with movement were consistently lower than those without movement. The percentage of problems correctly identified is shown in Figure 18. Again, a clear difference can be seen between animations without movement and animations with movement Problems were identified correctly just better than 50% of the time when the animation type was C, I, or CI, whereas the percent correct was 80% or better for M, MC, MI, and MCI.

Table 14. Problem Identification Time Means and Standard Deviations.

Statistic	C	I	CI	M	MC	MI	MCI
Mean	47	48	46	36	30	33	34
St Dev	15	14	18	15	13	15	16

Subject performance based on animation type was also examined by analyzing the data when the problems were correctly identified and incorrectly identified. Given that the problem was correctly identified, the animations with movement (M, MC, MI, and MCI) overall had a higher number of correct identifications and a lower average time in which those problems were recognized while the animations without movement (C, I, and CI) showed just the opposite. Figure 19 shows the results graphically. The graph for the incorrect identifications is shown in Figure 20. Again, with regards to number incorrect, the subjects performed better when the animations had movement. However, the average time of an incorrect identification was similar for all the animation types with M the highest.

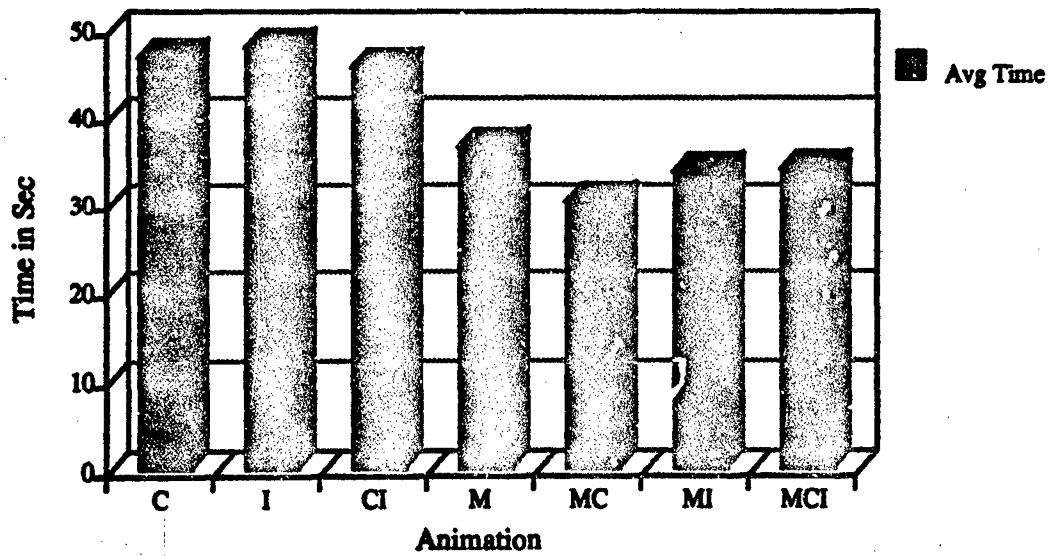


Figure 17. Average Time to Identify Potential Problem.

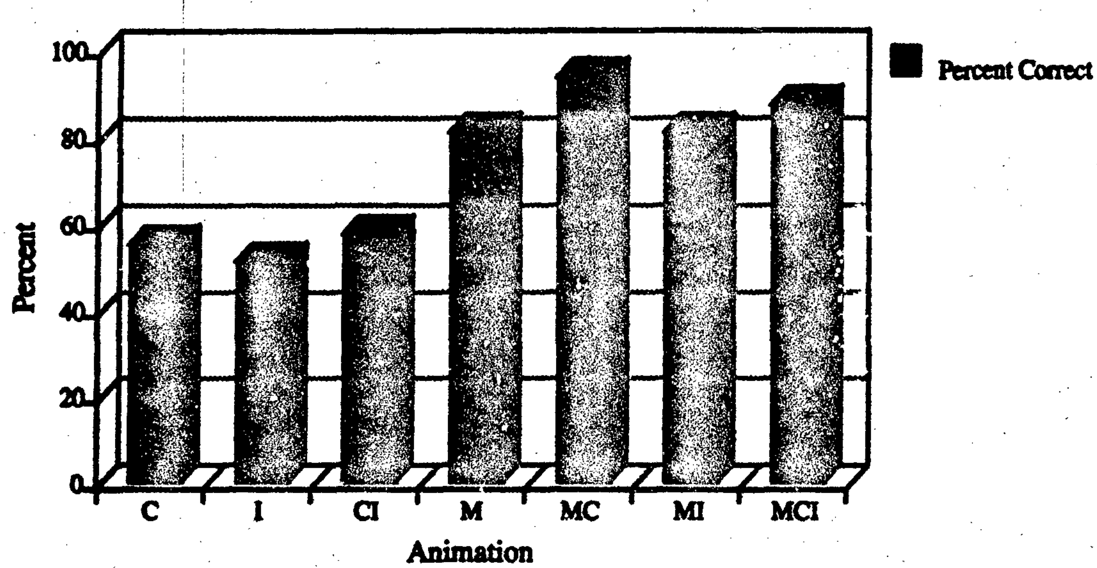


Figure 18. Percentage of Problems Correctly Identified.

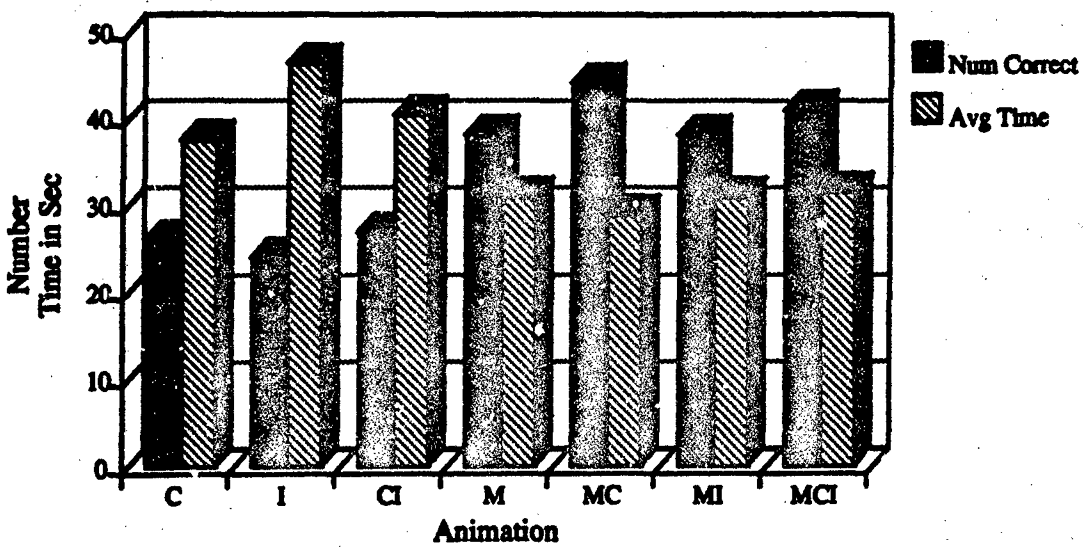


Figure 19. Number and Average Time of Problems Correctly Identified.

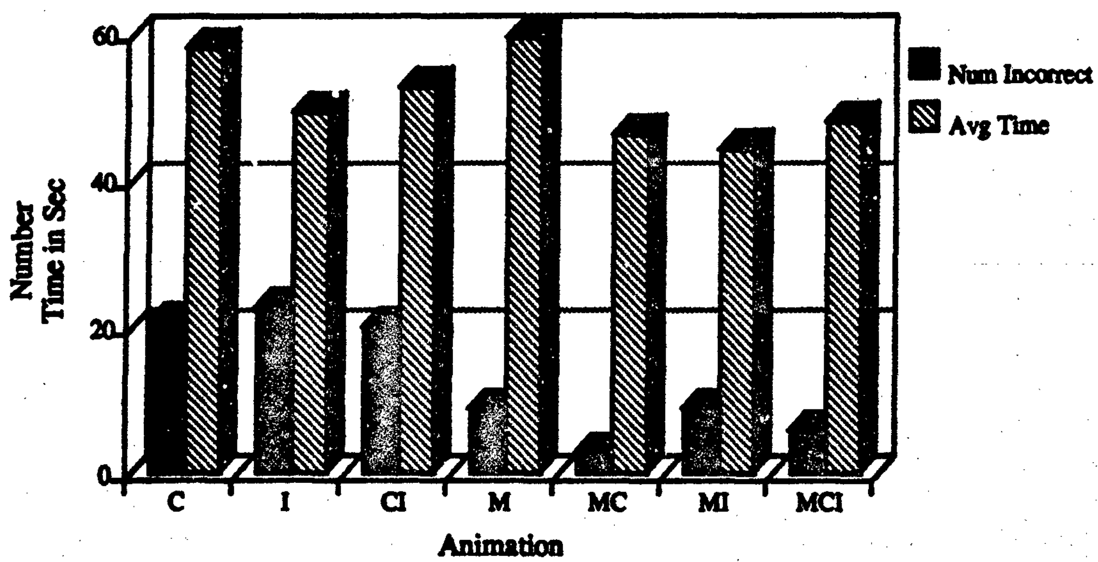


Figure 20. Number and Average Time of Problems Incorrectly Identified.

Another question of interest was how many subjects responded before and after the animations ended. Since the animations ran for 60 seconds, the issue was how many subjects responded within 60 seconds. Figure 21 shows the results. C, I, and CI were about even. About half responded within 60 seconds and about half after 60 seconds. M, MC, MI, and MCI were quite different. There were significantly more responses within 60 seconds than after 60 seconds.

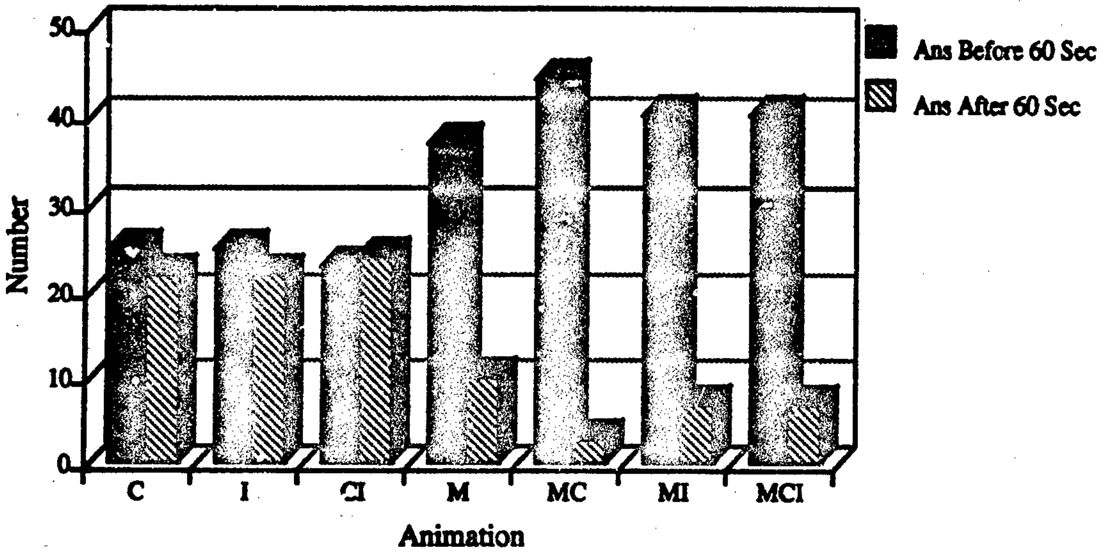


Figure 21. Number of Problem Identifications Before and After 60 Seconds.

Therefore, when the animations contained movement, the subjects performed better than when the animations did not contain movement. Unlike the AHP ratings, though, there did not seem to be any significant difference between M, MC, MI, and MCI.

Animation Perspective ANOVA. This Analysis of Variance assumed the same model as the AHP ANOVA, which is given in Equation (2). In this case the times Y_{ij} are assumed to be independent and normally distributed, with mean and constant variance

$$E\{Y_{ij}\} = \mu + T_i + B_j$$
$$\sigma^2\{Y_{ij}\} = \sigma^2$$

As before, each animation type was a treatment ($i = 1, \ldots, 7$), and each subject was a block ($j = 1, \ldots, 47$). So this model determined if there were differences in subject performance given an animation type and differences in performance between subjects. Table 15 shows the ANOVA table.

Table 15. ANOVA Table for Problem Identification Times from Animation Perspective.

Source	DF	SS	MS	F	P
Treatment	6	15437.2	2572.87	14.92	0.0000
Block	46	26524.0	576.61	3.34	0.0000
Error	276	47596.4	172.45		
Total	328	89557.7			
Grand Average	1	504500			

The ANOVA table shows that there is a difference in subject performance between animation types, and there is a difference in performance between subjects. That is, there is a block effect. This was expected since there was a variety of subjects, each with a different idea of what constitutes an efficiency problem. Also, some were familiar with simulation modeling and animation and some were not. A Tukey test for additivity was performed to test for interaction effects between animation type and subject. Since the null hypothesis of the test is no interaction effects (treatment and block are additive), the F value of 0.05 and the P value of 0.8308 indicated there was no animation type and subject interaction effects. Table 16 gives the complete results of the test.

Table 16. Tukey's 1 Degree of Freedom Test For Additivity.

Source	DF	SS	F	P
Nonadditivity	1	7.91337	0.05	0.8308
Remainder	275	47588.5		

Figure 22 shows a normal probability plot. The residuals were plotted against the expected residuals under normality to check the normality assumption. The almost straight line indicates that the normality assumption is valid. Another requirement for ANOVA is equal variances for the animation types (treatments). The residuals are plotted against the animation types in Figure 23. The plot seems to show equal variances, and there is a good scattering of the residuals in Figure 24, which shows the residuals versus the subjects (blocks). Since the block variable could not be dropped because of the block effect, a Bartlett's test of equal variances could not be performed.

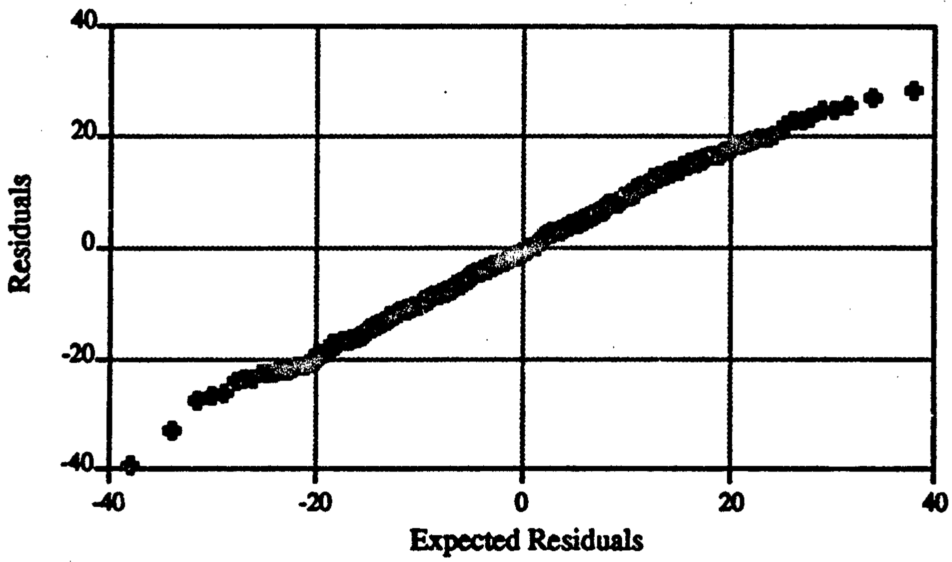


Figure 22. Normal Probability Plot of Residuals vs Expected Residuals.

The Tukey method of multiple comparisons was used to determine if the differences between the mean problem identification times of each animation type were statistically significant. The means were tested at a family α level of 0.1. The differences between the mean problem identification times of the animations without movement (C, I, and CI) were determined not to be statistically significant. That is, at the 90% confidence level the means were considered to be the same. Also, the differences between the mean problem identification times for the animation types with movement (M, MC, MI, and MCI) were not statistically different from each other; however they were statistically different from the group of animation types without movement. Thus, the test determined that there were two groups: (MCI, MI, MC, M) and (C, I, CI).

Animation Perspective Cluster Analysis. The cluster analysis techniques that were used with the AHP ratings were also used here: Complete Linkage, Average Linkage,

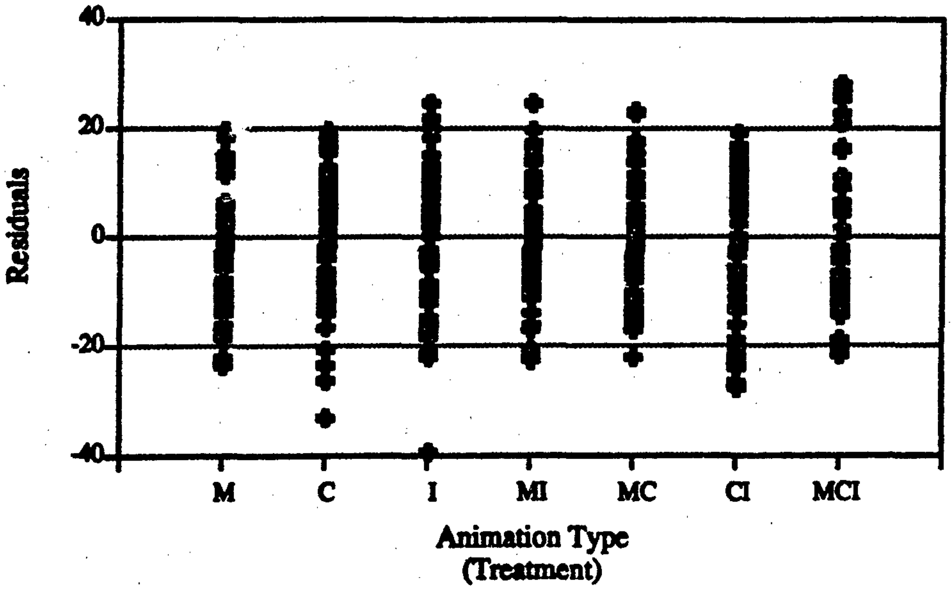


Figure 23. Scatter Plot of Residuals vs Animation Type (Treatment).

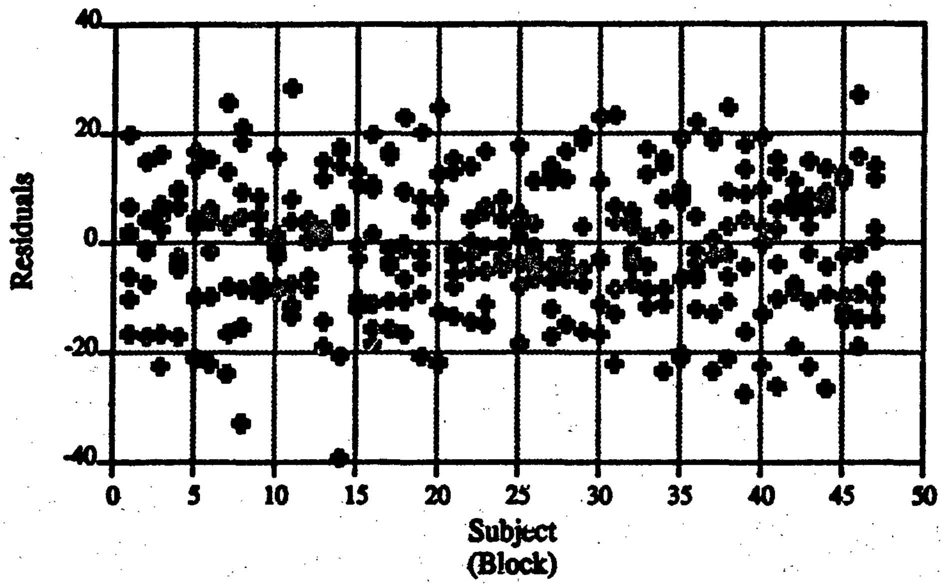


Figure 24. Scatter Plot of Residuals vs Subject (Block).

and Ward's Minimum Variance. All three techniques clustered the animations with movement first. M and MCI were clustered, as were MI and MC. Then all four were clustered. C and CI were clustered, then joined with I. This pattern was the same for all the techniques. Table 17 shows the clusters and the distance for the complete linkage technique, and Figure 25 is the dendogram for the complete linkage. Table 18 and Figure 26 show the same for the average linkage technique, and Table 19 and Figure 27 gives the results for Ward's minimum variance technique.

The cluster analysis yielded results similar to Tukey's method of multiple comparisons. There was very little difference in subject performance for animations without movement, and very little difference in subject performance for animations with movement. That is, movement significantly increased the subjects ability to discern the operation of the system. Also, adding color, detailed icons, or both did not increase the subject's ability to understand the operation of the model, as measured by the average problem identification time. So the cluster analysis confirmed that all of the animation types without movement were more difficult to interpret than the animations with movement, and movement was the most important communication tool.

Table 17. Complete Linkage Cluster Analysis Results for Problem ID Times from Animation Perspective.

Animations Joined	Distance
M, MCI	0.749877
MI, MC	0.801696
C, CI	0.887343
M, MCI, MI, MC	0.946333
C, CI, I	0.974192
M, MCI, MI, MC, C, I, CI	1.224518

Problems Results From Scenario Perspective.

Summary Statistics. As with the problem results from the animation perspective, two data items were examined: problem identification time and problem identification accuracy. Whereas the problem data looked at from the animation perspective measured

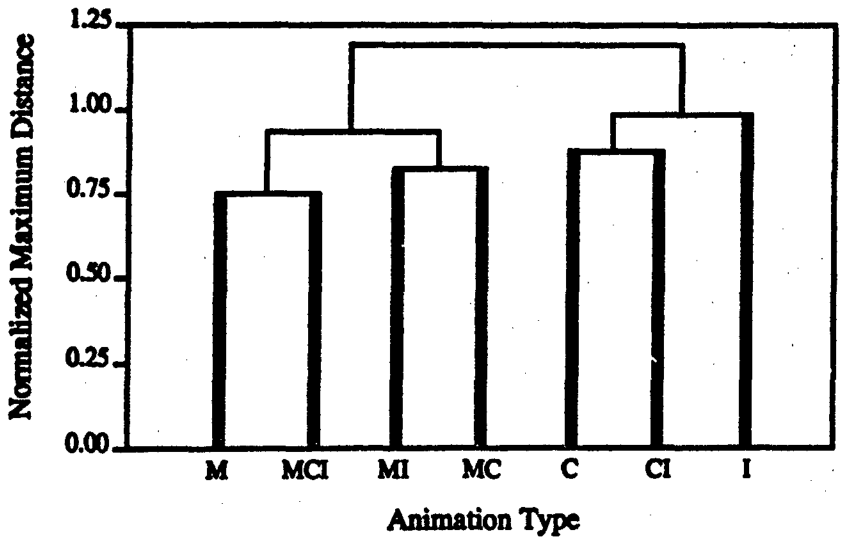


Figure 25. Complete Linkage Dendogram for Problem ID Times from Animation Perspective.

Table 18. Average Linkage Cluster Analysis Results for Problem ID Times from Animation Perspective.

Animations Joined	Distance
M, MCI	0.743121
MI, MC	0.794472
M, MCI, MI, MC	0.870167
C, CI	0.879348
C, CI, I	0.957954
M, MCI, MI, MC, C, I, CI	1.087015

Table 19. Ward's Minimum Variance Cluster Analysis Results for Problem ID Times from Animation Perspective.

Animations Joined	Distance
M, MCI	0.092038
MI, MC	0.105198
C, CI	0.128875
M, MCI, MI, MC	0.153779
C, CI, I	0.160969
M, MCI, MI, MC, C, I, CI	0.359141

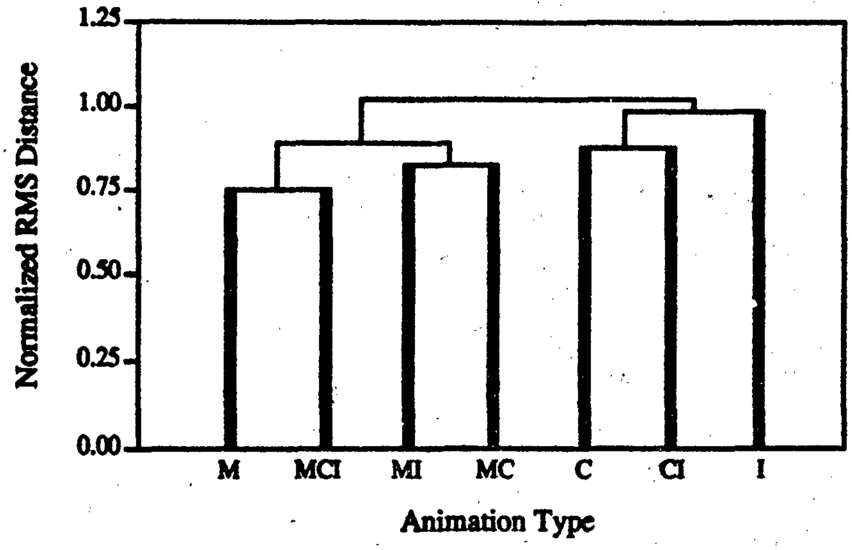


Figure 26. Average Linkage Dendogram for Problem ID Times from Animation Perspective.

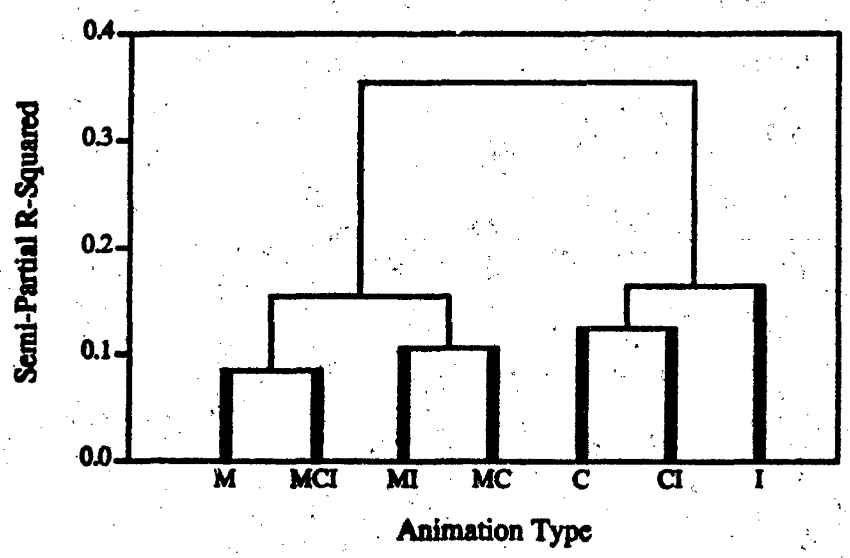


Figure 27. Ward's Minimum Variance Dendogram for Problem ID Times from Animation Perspective.

the performance of the subjects given an animation type, the problem data analyzed from the scenario perspective shows which scenarios were most recognizable to the subjects. Figure 28 shows the average time (in seconds) that a potential problem was identified, and Table 20 gives the numbers. Three of the scenarios (FL, LDER, and SLTK) appear to have higher times than the others. However, only FL stands out when the percentage of problems correctly identified is examined. These are shown in Figure 29.

Table 20. Problem Identification Time Means and Standard Deviations from Scenario Perspective.

Statistic	FL	LDER	SLTK	ST	SL	FT	SLLD
Mean	46	42	41	35	37	36	36
St Dev	16	17	14	20	17	16	12

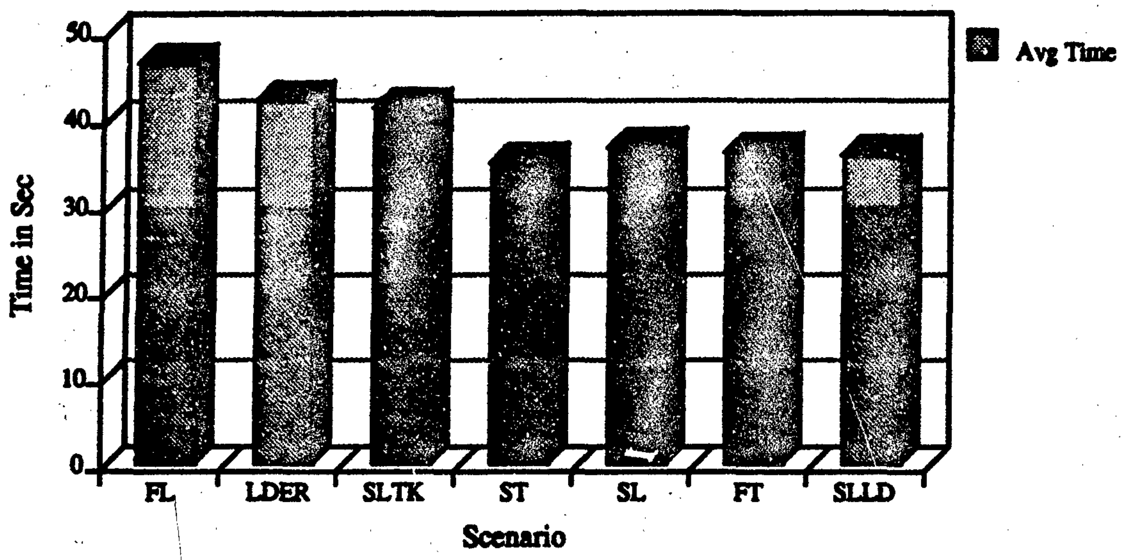


Figure 28. Average Time to Identify Potential Problem from Scenario Perspective.

Scenario Perspective ANOVA. This Analysis of Variance assumed the same model as the AHP ANOVA and animation perspective ANOVA, which is given in Equation (2). In this case the times Y_{ij} are assumed to be independent and normally distributed,

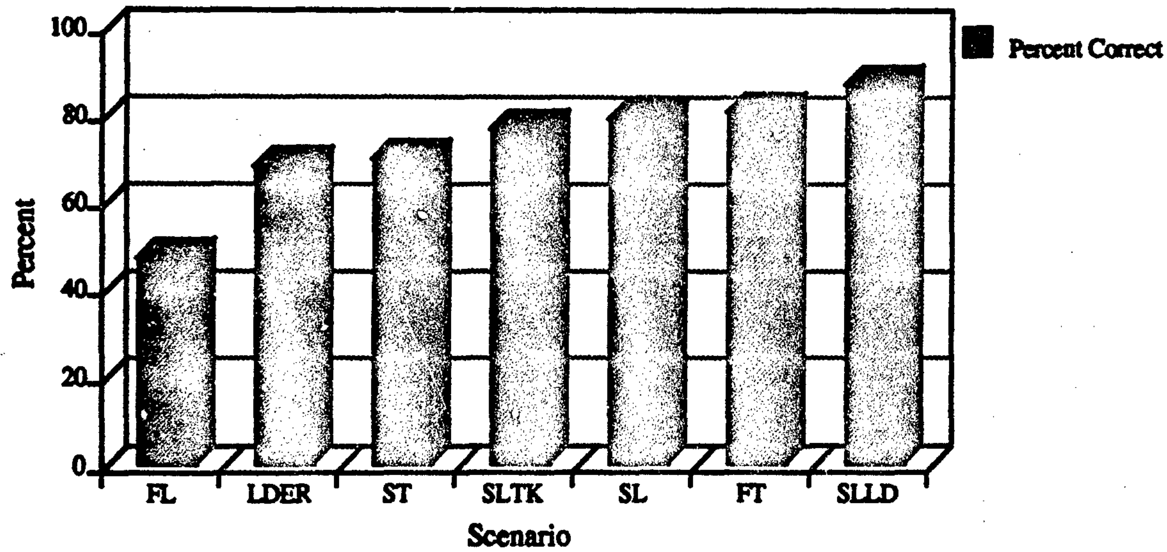


Figure 29. Percentage of Problems Correctly Identified from Scenario Perspective.

with mean and constant variance

$$
\begin{aligned}
E\{Y_{ij}\} &= \mu + T_i + B_j \\
\sigma^2\{Y_{ij}\} &= \sigma^2
\end{aligned}
$$

In this case, however, each scenario was a treatment ($i = 1, \ldots, 7$), and each subject was a block ($j = 1, \ldots, 47$). So this model determined if there were differences in subject performance among the scenarios and differences in performance between subjects. The purpose of this ANOVA was to determine if there was a statistical difference in subject performance between the scenarios. Table 21 is the ANOVA table.

Table 21. ANOVA Table for Problem Identification Times from Scenario Perspective.

Source	DF	SS	MS	F	P
Treatment	6	5096.44	849.407	4.00	0.0008
Block	46	26243.0	570.500	2.68	0.0000
Error	276	58650.9	212.503		
Total	328	89990.4			
Grand Average	1	502700			

The ANOVA table shows that there is a difference in subject performance between scenarios, and there is a difference in performance between subjects. That is, there is a block effect. A Tukey test for additivity was performed to test for interaction effects between scenario and subject. Since the null hypothesis of the test is no interaction effects (treatment and block are additive), the F value of 0.14 and the P value of 0.7114 indicated there was no scenario and subject interaction effects. Table 22 gives the complete results of the test.

Table 22. Tukey's 1 Degree of Freedom Test For Additivity.

Source	DF	SS	F	P
Nonadditivity	1	29.246	0.14	0.7114
Remainder	275	58621.7		

Figure 30 shows a normal probability plot. The residuals were plotted against the expected residuals under normality to check the normality assumption. The almost straight line indicates that the normality assumption is valid. Another requirement for ANOVA is equal variances for the scenarios (treatments). The residuals are plotted against the scenarios in Figure 31. The plot seems to show equal variances, and there is a good scattering of the residuals in Figure 32, which shows the residuals versus the subjects (blocks). As with the ANOVA for the problem identification times from the animation perspective, a Bartlett's test of equal variances could not be performed because of the block effect.

The Tukey method of multiple comparisons was used to determine if the differences between the mean problem identification times of each scenario were statistically significant. The means were tested at a family α level of 0.1. The differences between the mean problem identification times for FL, LDER, and SLTK were determined not to be statistically significant. That is, at the 90% confidence level the means were considered to be the same. Also, the differences between the mean problem identification times for SL, FT, SLLD, and ST were not statistically different from each other. Thus, the test determined that there were two groups: (FL, LDER, SLTK) and (SL, FT, SLLD, ST). This grouping had been suggested by Figure 28.

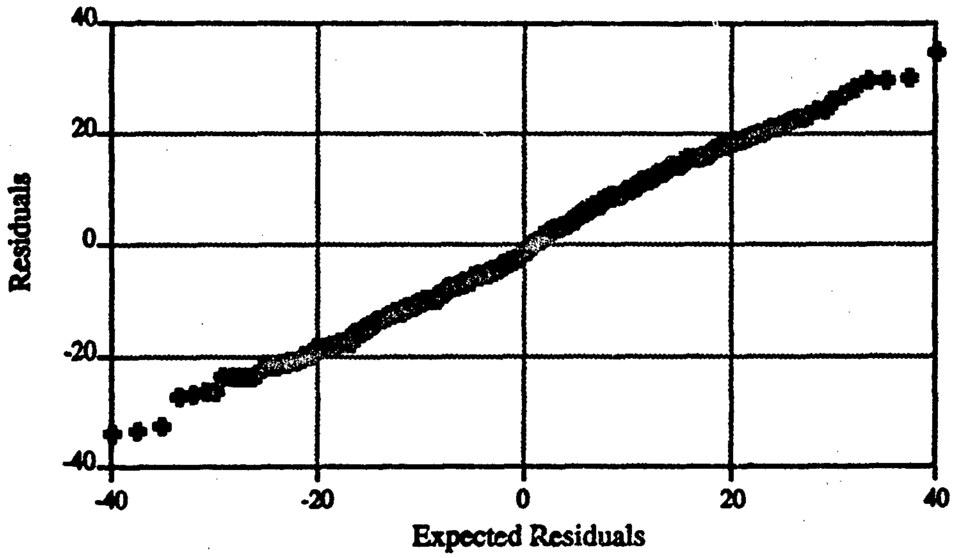


Figure 30. Normal Probability Plot of Residuals vs Expected Residuals.

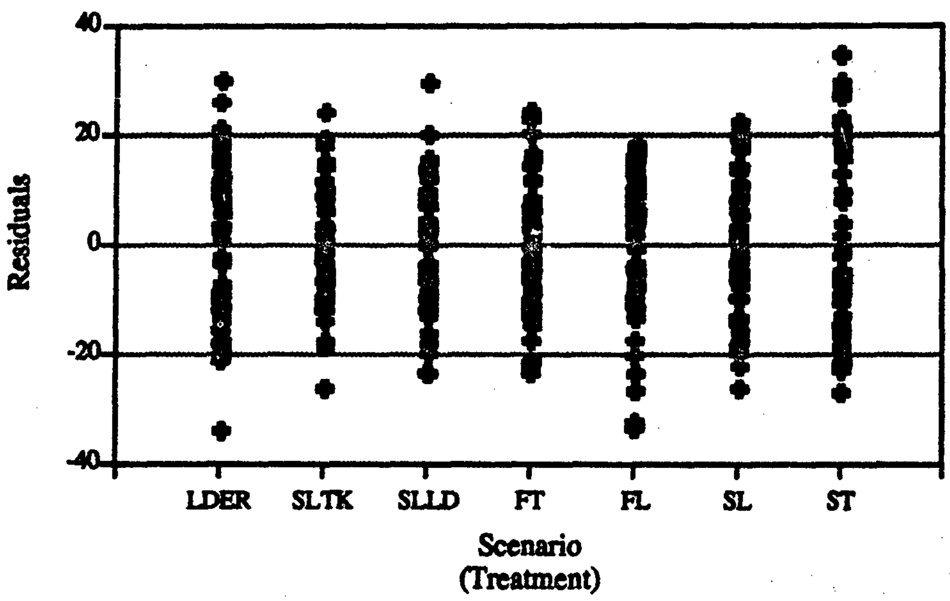


Figure 31. Scatter Plot of Residuals vs Scenario (Treatment).

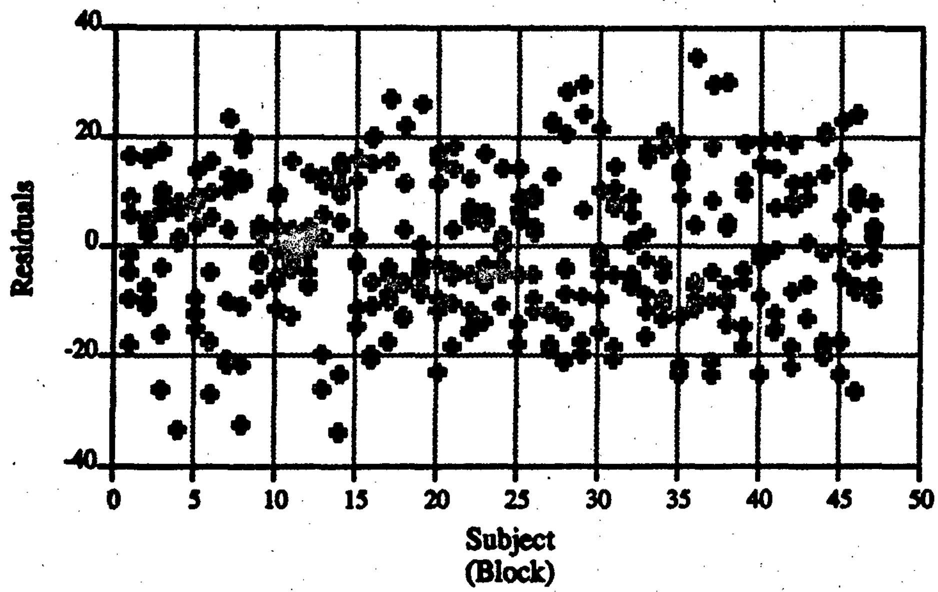


Figure 32. Scatter Plot of Residuals vs Subject (Block).

The higher average problem identification times for FL, LDER, and SLTK, can be explained. The times for FL were higher on the average because this scenario was an under utilization problem. In FL the loaders were faster, which caused more loader idle time. This caused a higher average problem identification time (and lower problem identification accuracy) because fewer subjects recognized idle loaders as a problem. Many thought there was no problem with FL. In FT the trucks were faster, which caused more truck idle time; however, most recognized this under utilization. This is probably because the trucks waited in line for loads. So two or more trucks waiting to load (in the middle of the computer screen) were more obvious than two loaders (at the bottom of the computer screen) waiting in separate locations. Another possibility is the subjects did not recognize the idle loaders because they were not truly "system experts". LDER had a higher average problem identification time because no problem could be seen in the animations without movement (C, I, and CI). The only problem in LDER was a backup of loads, which was not shown in C, I, and CI. Thus, usually the full 60 seconds would elapse without a problem identification. The load queue was not included in the icon-only animation (I) because the researcher was uncertain how to show the load queue without movement. It was not shown in the color-only animation (C) because that would have required two different color keys on the screen. The reasons for not including the load queue in C and I also apply to CI. In retrospect, however, there should have been some representation of the load queue in the animations without movement. The average problem identification time for SLTK was higher because the backup of loads caused by the one slow truck showed up later in the animation. Therefore, the problem of backup of loads could not be identified as early as problems in the other scenarios. The problem of having one slow truck could have been identified early, though. It is interesting that very few subjects recognized one truck being slower than the others as a problem. Most subjects focused on the loading area only (for animations with movement) and did not notice that one truck was consistently being passed. That is, one truck was consistently taking much longer to haul, dump, and return.

Therefore, looking at the data from the scenario perspective suggested several things. First, animation may not tell the whole story. Also, several animations should be developed that show the system from different perspectives. For instance, with LDER, the animations without movement did not show the problem with the system (buildup of loads), and with FL and SLTK, some areas of the computer screen were not given as much attention as others. The latter implies breaking up the system into several views. That is, view animations that zoom in on certain areas as well as view animations that give an overview of the complete system.

Comparison of Subjective and Objective Results

For the animations without movement (C, I, and CI) the subjective and objective results match well. These animation types were rated as poor communicators and the subjects' performance when viewing the animations without movement agreed with those ratings. There was a clear distinction between the animations with movement (M, MC, MI, and MCI) and the animations without movement in both the subjective and objective results. M, MC, MI, and MCI were much preferred over C, I, and CI, and the subjects performed significantly better when viewing the animations with movement. However, given that there was movement in the animation, the subjective and objective results differed. The subjects preferred movement as the primary communication tool, but the subjects rated communication higher as color and detailed icons were added to the animations. MCI was selected 79% of the time as the animation that communicated the best, and MCI was rated the highest in the pairwise comparisons. The objective results showed, though, that there was no difference in subject performance for any of the animations with movement. Therefore, even though the subjects preferred a more detailed animation, movement was the primary factor.

Table 23 summarizes the comparison of the subjective and objective results and summarizes the result of this chapter by showing the animation groupings from the Tukey method of multiple comparisons. C, I, and CI were collectively judged to communicate poorly, and the subjects did not perform well when viewing these animation types. Although movement was considered by the subjects to be important, their preferences increased as color and detailed icons were added. With regard to subject performance, however, there was no difference between the animations with movement (M, MI, MC, MCI).

Table 23. Animation Groupings for Subjective and Objective Evaluations.

Evaluation	Groups Formed
Subjective	C, I, CI
	M
	MC
	MI
	MCI
Objective	C, I, CI
	M, MC, MI, MCI

V. Conclusions and Recommendations

Conclusions

Summary. As stated in Chapter 1, animation is a communication tool that can help in establishing a model's face validity. Since communicating the operation of a simulation model to the system experts is necessary to determine face validity, this research looked at three aspects of animation (color, detail of icons, and movement) to determine which ones were the most useful for communicating the operation of a simulation model. This ability to communicate was measured both subjectively and objectively. The subjective measures were a selection of "best" and "worst" animation types where "best" and "worst" referred to how well an animation communicated, and a pairwise comparison of the animation types which resulted in preference ratings for each animation. There were seven different scenarios containing various problems with the system. The objective measures were subject problem identification accuracy and time delay of problem identification.

Results. The results showed that movement was the most important aspect of animation. Animations with movement were much preferred over animations without movement, and the subjects identified problems more accurately in less time when viewing animations with movement than animations without movement. Whereas there were differences among the AHP ratings for M, MC, MI, and MCI, there was no difference among these animations with regards to subject performance. Further examination of the AHP ratings revealed that, given movement, detailed icons were preferred over color. Also, those who liked detailed icons did not like color, and those who liked color did not like detailed icons. However, most preferred having movement, color, and detailed icons.

The potential for bias in the results should be mentioned here. Looking back, it is possible that movement was shown to be most important because of the scenarios used. Since the problems associated with each scenario were primarily utilization problems, they might have shown up easier with movement than with color or detailed icons. Other types of problems (such as a truck leaving empty) might have been seen more quickly with color or detailed icons than with movement. Nevertheless, some or all of the problems associated with each scenario (except for the LDER scenario as discussed in Chapter 4) could be identified in each animation type.

Additional Observations. The simulation model used in this experiment was chosen because it was simple and concise. However, the model was not designed with animation in mind. Several modifications and simplifications were made to the model so that is could be animated. Looking back, a couple of the modifications might not have been needed if the researcher had had more experience with animation and with Proof. Nevertheless, some model modifications would still have been required. The primary addition to the model was the ability to keep track of the trucks and loaders. Creating the animation trace files would have been easier if each separate movement of an entity had been explicitly modeled. As described in Chapter 3, the loading times had to be artificially divided to account for the various movements required by the loaders. Therefore, if a modeler anticipates animating a model, this should be kept in mind when designing and coding the model. The modeler should make certain, though, that the system being modeled determines the model design and not animation considerations.

Finally, even though the initial aim of this study was to examine animation's role in establishing face validity, the contribution of animation to face validity was not what was actually measured. The simulation model that was used was assumed to be valid; therefore, the subjects could not judge the model's face validity. In the context of face validity, the research examined which aspects of animation best communicated the operation of the model. So the result that using movement in animations is important applies, not only to face validity, but also to other validation and verification techniques. In addition, this result applies to any other areas in which animation could be used, such as communicating the model to a decision maker.

Recommendations for Further Study

Several aspects of animation that were not considered in this study (such as graphs and speed of animation) were mentioned in Chapter 1. The aspects of animation that were not examined, plus what was learned during this research suggest the following studies:

- Repeat this research with a larger simulation model of an actual system. That is, investigate movement, color, and detail of icons with a model of a more complex, real world system. The model should have more simultaneous activities and a larger number of entities. Using this type of model would allow real "system experts" to rate the animations. Also, the results from this type of study would assist in determining whether the results of this study hold for a more realistic scenario.

- Investigate the use of graphs alone or in combination with other aspects of animation. This research could address such questions as:

  1. Is a graph showing queue status necessary when the actual entities can be seen waiting?

  2. What information can be displayed with graphs that can not be displayed with the aspects of animation investigated here?

  3. Is the unique information displayed by graphs critical?

  4. Do graphs improve communication or become distractions?

- Examine the impact of the viewing speed of the animations. Many times during the viewing of the animations without movement, the subjects commented that the icons or colors were changing too fast. How would have the subjects performed if the animations without movement had been slower? Is there an optimal viewing speed, and can it be determined?

- Study the usefulness of color in communicating when the colors have well known meanings in the context of the system being modeled. For instance, in this study, the colors were assigned arbitrarily. White represented an idle truck or loader, and red represented a loaded truck or loader. These were subjective color assignments. What if, in the system modeled, red meant stop (as on a traffic light or stop sign), or red meant hot? When there is meaning to the colors, the importance of color as a communication tool might increase.

Appendix A. *Data Collection Forms*

This appendix shows the data collection forms used. Page 61 displays the form used to document which scenario and animation was viewed, the time required to identify the problem, and the problem identified. The pairwise comparison data collection instructions, examples, and form follow on page 62 through page 64. Page 65 shows the pairwise matrix form. The model description ends this appendix and begin on page 66.

Scenario/Animation Viewing

Name: __________ **Date:** __________ **Run:** __________

Student _ **Faculty** _ **Other** _

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Scenario: ___ **Animation:** ___ **Problem ID Time:** ___

Problem Observed: ______________________________

Best Animation for Face Validity: ______

Worst Animation for Face Validity: ______

Animation Pairwise Comparison

The pairwise comparison of the animations allows you to rate an animation's ability to communicate the model's operation against the ability of another animation. This allows the relative merits of the animations to be quantified. Since there are seven animations, you will perform 21 pairwise comparisons. After you have completed the first few comparisons, I will briefly interrupt you. This is so I can relate back to you one or two of your responses to verify that I have communicated the instructions properly. Feel free to go back and erase a previous response if you think it needs adjusting. The order of your responses is not important. What I need is a completed form that reflects your best comparison of the animations.

As a reminder, the animation types are:

- M - Movement. Simple icons that move but do not change level of detail or color.
- I - Icon. Icons change level of detail but do not move or change color.
- C - Color. Simple icons that change color but do not move or change level of detail.
- MI - Movement and Icon. Icons move and change level of detail but do not change color.
- MC - Movement and Color. Simple icons that move and change color but do not change level of detail.
- CI - Color and Icon. Icons change color and level of detail but do not move.
- MCI - Movement, Color, and Icon. Icons move and change color and level of detail.

The pictures of each type of animation are available for you to look at while you are performing the comparisons. The next page gives some pairwise comparison examples, and the final page is the pairwise comparison form.

Pairwise Comparison Examples

Animation A and Animation B have equal contribution to face validity.

>>>> >>> >> > = < << <<< <<<<

A _ _ _ _ _ _ _ _ | X | _ _ _ _ _ _ _ _ **B**

Animation A contributes slightly more to face validity than Animation B.

>>>> >>> >> > = < << <<< <<<<

A _ _ _ _ _ _ X _ | _ | _ _ _ _ _ _ _ _ **B**

Animation A contributes somewhat more to face validity than Animation B.

>>>> >>> >> > = < << <<< <<<<

A _ _ _ X _ _ _ _ | _ | _ _ _ _ _ _ _ _ **B**

Animation A contributes much more to face validity than Animation B.

>>>> >>> >> > = < << <<< <<<<

A X _ _ _ _ _ _ _ | _ | _ _ _ _ _ _ _ _ **B**

Animation B contributes slightly more to face validity than Animation A.

>>>> >>> >> > = < << <<< <<<<

A _ _ _ _ _ _ _ _ | _ | _ X _ _ _ _ _ _ **B**

Highest Contribution to Face Validity

Name: ____________ **Date:** ____________ **Run:** ____________

Student _ **Faculty** _ **Other** _

	>>>>	>>>	>>	>	=	<	<<	<<<	<<<<	
M	— —	— —	— —	— —	—	— —	— —	— —	— —	C
M	— —	— —	— —	— —	—	— —	— —	— —	— —	I
M	— —	— —	— —	— —	—	— —	— —	— —	— —	MI
M	— —	— —	— —	— —	—	— —	— —	— —	— —	MC
M	— —	— —	— —	— —	—	— —	— —	— —	— —	CI
M	— —	— —	— —	— —	—	— —	— —	— —	— —	MCI
C	— —	— —	— —	— —	—	— —	— —	— —	— —	I
C	— —	— —	— —	— —	—	— —	— —	— —	— —	MI
C	— —	— —	— —	— —	—	— —	— —	— —	— —	MC
C	— —	— —	— —	— —	—	— —	— —	— —	— —	CI
C	— —	— —	— —	— —	—	— —	— —	— —	— —	MCI
I	— —	— —	— —	— —	—	— —	— —	— —	— —	MI
I	— —	— —	— —	— —	—	— —	— —	— —	— —	MC
I	— —	— —	— —	— —	—	— —	— —	— —	— —	CI
I	— —	— —	— —	— —	—	— —	— —	— —	— —	MCI
MI	— —	— —	— —	— —	—	— —	— —	— —	— —	MC
MI	— —	— —	— —	— —	—	— —	— —	— —	— —	CI
MI	— —	— —	— —	— —	—	— —	— —	— —	— —	MCI
MC	— —	— —	— —	— —	—	— —	— —	— —	— —	CI
MC	— —	— —	— —	— —	—	— —	— —	— —	— —	MCI
CI	— —	— —	— —	— —	—	— —	— —	— —	— —	MCI

Pairwise Matrix

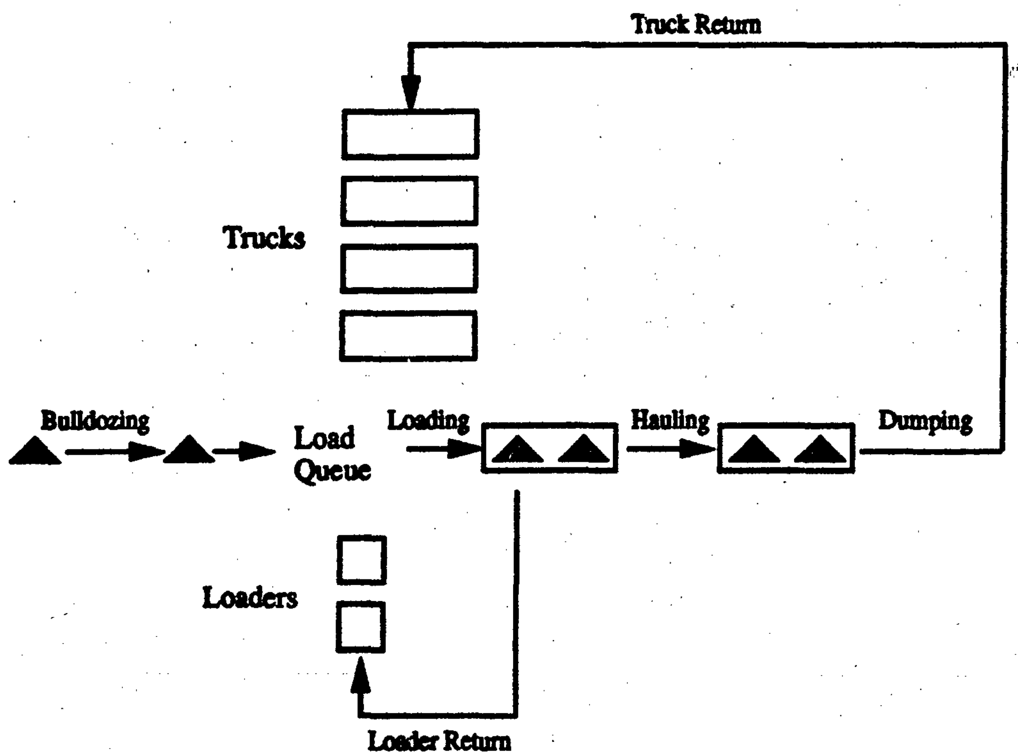
Name: ______________ **Date:** ______________ **Run:** ______

	M	C	I	MI	MC	CI	MCI	Ratings
M	1							______
C		1						______
I			1					______
MI				1				______
MC					1			______
CI						1		______
MCI							1	______

S^2 = ________ P or F

A Truck Hauling Situation

The system modeled consists of one bulldozer, four trucks, and two man-machine loaders. The bulldozer stockpiles material for the loaders. Two piles of material must be stocked prior to the initiation of any load operation. In addition to the two loads of material, a loader and an unloaded truck must be available before the loading operation can begin. There is room in the loading area for only one truck and one loader. After a truck is loaded, it is hauled and then dumped. It must be returned before the truck is available for further loading. Also, the dumping area can only accommodate one truck. Following a loading operation, the loader must return to its idle position before it is available to begin loading again. The diagram below shows the flow of the system.



You, as the owner of the trucks, loaders, and bulldozer, requested a simulation to study the efficiency of your operation. My job as the simulation model builder is to communicate to you the operation of the model so that you can determine whether the model accurately reflects your application. This procedure is establishing the face validity of the model. I have decided to use animation to relate how your system is modeled. You will view seven different animations of the model. While viewing the animations, try to determine whether there is a problem with the system. Any particular animation may or may not exhibit a problem. Problems could include over or under utilization of the trucks, loaders, or bulldozer; a large build-up of loads; an inadequate number of trucks, loaders, and bulldozers; or a combination of these. After all the animations have been viewed, you will perform a pairwise comparison of each type of animation. That is, you will compare how each type of animation communicates the operation of the system (face validity) with every other type of animation.

The types of animation are:

- M - Movement. simple icons that move but do not change level of detail or color.
- I - Icon. Icons change in level of detail but do not move or change color.
- C - Color. Simple icons that change color but do not move or change in level of detail.
- MI - Movement and Icon. Icons move and change in level of detail but do not change color.
- MC - Movement and Color. Simple icons that move and change color but do not change in level of detail.
- CI - Color and Icon. Icons change color and level of detail but do not move.
- MCI - Movement, Color, and Icon. Icons move, change color, and change level of detail.

The above abbreviations will be used to identify each animation. Using pictures I will describe the layout of each animation. When viewing the animations, speak out as soon as you think you have discovered a problem (such as saying "I got it"). Feel free to ask questions before you begin viewing the animations. Let me know when you are ready to start.

Appendix B. *SAS Output*

This appendix contains the SAS output used in the analysis. The SAS command file begins on page 69, and the output listing begins on page 71. The data sets used are:

- GMEANS - AHP ratings for each animation type.
- GMEANST - AHP ratings transposed. That is, instead of the ratings for each animation type being in columns, the ratings are in rows. This transformation was necessary for the cluster analysis.
- PROBS - Problem data from animation perspective. Each animation type had an accuracy variable and a time variable. For instance, for M, MP contained the problem accuracy data (0 for incorrect, 1 for correct), and MT contained the problem identification times.
- RPROBS - Problem data for the correct identifications only. D1 through D7 were used for the problem identification accuracy data because that data was not needed (all were 1).
- WPROBS - Problem data for the incorrect identifications only. D8 through D14 were used for the problem identification accuracy data because that data was not needed (all were 0).
- INTIME - Problem data in which the subject responded before 60 seconds had elapsed. For example, MNA contains the movement animation (M) problem accuracy data, and MNT contains the problem time data for M.
- OUTTIME - Problem data in which the subject responded before 60 seconds had elapsed. D15 through D21 were used for the problem identification time data because that data was not needed (all were 60).
- PROBST - Problem identification times transposed for cluster analysis.
- SCENARIO - Problem data from scenario perspective. Each scenario had an accuracy variable and a time variable. The variables follow the same pattern as PROBS. For example, LDERP contained the problem accuracy data for the LDER scenario, and LDERT contained the problem identification times for the LDER scenario.

```
OPTIONS LINESIZE=78;
DATA GMEANS;
  INPUT M C I MI MC CI MCI;
  %INCLUDE MEANS;
DATA GMEANST;
  INPUT P1-P47;
  %INCLUDE MEANST;
DATA PROBS;
  INPUT MP MT CP CT IP IT MIP MIT MCP MCT CIP CIT MCIP MCIT;
  %INCLUDE PROBS;
DATA RPROBS;
  INPUT D1 MRP D2 CRP D3 IRP D4 MIRP D5 MCRP D6 CIRP D7 MCIRP;
  %INCLUDE RPROBS;
DATA WPROBS;
  INPUT D8 MWP D9 CWP D10 IWP D11 MIWP D12 MCWP D13 CIWP D14 MCIWP;
  %INCLUDE WPROBS;
DATA INTIME;
  INPUT MNA MNT CNA CNT INA INT MINA MINT MCNA MCNT CINA CINT MCINA MCINT;
  %INCLUDE INTIME;
DATA OUTTIME;
  INPUT MOA D15 COA D16 IOA D17 MIOA D18 MCOA D19 CIOA D20 MCIOA D21;
  %INCLUDE OUTTIME;
DATA PROBST;
  INPUT T1-T47;
  %INCLUDE PROBST;
DATA SCENARIO;
  INPUT LDERP LDERT SLTKP SLTKT SLLDP SLLDT FTP FTT FLP FLT SLP SLT STP STT;
  %INCLUDE SCENDAT;
PROC PRINT DATA=GMEANS;
PROC MEANS DATA=GMEANS;
PROC FACTOR DATA=GMEANS NFACTORS=3 METHOD=PRIN COV ROTATE=VARIMAX
            OUT=SCORES;
  VAR M C I MI MC CI MCI;
PROC PRINT DATA=SCORES;
  VAR FACTOR1 FACTOR2 FACTOR3;
PROC PLOT DATA=SCORES;
  PLOT FACTOR2*FACTOR1 FACTOR3*FACTOR1 FACTOR3*FACTOR2;
PROC CLUSTER DATA=GMEANST METHOD=COMPLETE;
PROC TREE;
PROC CLUSTER DATA=GMEANST METHOD=AVERAGE PSEUDO;
PROC TREE;
PROC CLUSTER DATA=GMEANST METHOD=WARD PSEUDO;
PROC TREE;
PROC PRINT DATA=PROBS;
  VAR MP MT CP CT IP IT MIP MIT MCP MCT CIP CIT MCIP MCIT;
```

```
PROC MEANS DATA=PROBS;
PROC PRINT DATA=RPROBS;
  VAR MRP CRP IRP MIRP MCRP CIRP MCIRP;
PROC MEANS DATA=RPROBS;
  VAR MRP CRP IRP MIRP MCRP CIRP MCIRP;
PROC PRINT DATA=WPROBS;
  VAR MWP CWP IWP MIWP MCWP CIWP MCIWP;
PROC MEANS DATA=WPROBS;
  VAR MWP CWP IWP MIWP MCWP CIWP MCIWP;
PROC PRINT DATA=INTIME;
PROC MEANS DATA=INTIME;
PROC PRINT DATA=OUTTIME;
  VAR MOA COA IOA MIOA MCOA CIOA MCIOA;
PROC MEANS DATA=OUTTIME;
  VAR MOA COA IOA MIOA MCOA CIOA MCIOA;
PROC CLUSTER DATA=PROBST METHOD=COMPLETE;
PROC TREE;
PROC CLUSTER DATA=PROBST METHOD=AVERAGE PSEUDO;
PROC TREE;
PROC CLUSTER DATA=PROBST METHOD=WARD PSEUDO;
PROC TREE;
PROC PRINT DATA=SCENARIO;
PROC MEANS DATA=SCENARIO;
```

OBS	M	C	I	MI	MC	CI	MCI
1	0.072	0.020	0.033	0.212	0.200	0.064	0.399
2	0.072	0.018	0.076	0.243	0.197	0.068	0.326
3	0.205	0.019	0.028	0.275	0.168	0.031	0.275
4	0.104	0.020	0.033	0.246	0.152	0.043	0.402
5	0.091	0.020	0.051	0.214	0.180	0.073	0.371
6	0.117	0.019	0.038	0.273	0.111	0.058	0.384
7	0.109	0.017	0.028	0.326	0.084	0.039	0.397
8	0.173	0.038	0.024	0.234	0.137	0.057	0.338
9	0.037	0.038	0.039	0.130	0.231	0.102	0.425
10	0.111	0.022	0.030	0.237	0.141	0.047	0.413
11	0.047	0.015	0.033	0.176	0.159	0.085	0.484
12	0.119	0.042	0.019	0.194	0.213	0.071	0.342
13	0.074	0.016	0.024	0.248	0.181	0.039	0.418
14	0.156	0.012	0.036	0.421	0.083	0.024	0.267
15	0.135	0.100	0.040	0.129	0.166	0.124	0.306
16	0.255	0.030	0.020	0.126	0.139	0.144	0.287
17	0.117	0.025	0.063	0.291	0.148	0.034	0.321
18	0.128	0.021	0.048	0.245	0.112	0.041	0.405
19	0.116	0.029	0.038	0.243	0.174	0.053	0.347
20	0.088	0.026	0.025	0.202	0.158	0.039	0.463
21	0.145	0.025	0.055	0.244	0.254	0.055	0.221
22	0.154	0.021	0.029	0.223	0.223	0.028	0.323
23	0.112	0.022	0.033	0.335	0.154	0.048	0.297
24	0.115	0.022	0.033	0.287	0.158	0.039	0.346
25	0.115	0.015	0.028	0.278	0.138	0.044	0.383
26	0.147	0.020	0.026	0.341	0.138	0.033	0.296
27	0.348	0.017	0.035	0.241	0.093	0.034	0.232
28	0.125	0.019	0.030	0.171	0.240	0.041	0.374
29	0.087	0.015	0.038	0.295	0.137	0.054	0.373
30	0.107	0.036	0.015	0.219	0.267	0.031	0.325
31	0.131	0.025	0.037	0.146	0.16[illegible]	0.064	0.434
32	0.095	0.029	0.066	0.285	0.154	0.070	0.301
33	0.033	0.054	0.021	0.260	0.238	0.061	0.332
34	0.072	0.114	0.024	0.065	0.272	0.093	0.360
35	0.078	0.023	0.029	0.154	0.165	0.078	0.473
36	0.105	0.045	0.039	0.213	0.162	0.058	0.378
37	0.142	0.047	0.040	0.242	0.131	0.058	0.339
38	0.114	0.018	0.030	0.250	0.150	0.043	0.395
39	0.156	0.023	0.023	0.360	0.104	0.046	0.288
40	0.088	0.025	0.022	0.131	0.226	0.043	0.465
41	0.070	0.042	0.025	0.152	0.262	0.063	0.385
42	0.123	0.015	0.034	0.280	0.148	0.041	0.359
43	0.153	0.046	0.062	0.282	0.139	0.082	0.235
44	0.097	0.020	0.044	0.164	0.193	0.067	0.415
45	0.099	0.020	0.038	0.291	0.144	0.054	0.354
46	0.280	0.024	0.049	0.179	0.216	0.036	0.216
47	0.123	0.019	0.030	0.246	0.155	0.037	0.389

Variable	N	Mean	Std Dev	Minimum	Maximum
M	47	0.1221277	0.0575680	0.0330000	0.3480000
C	47	0.0286809	0.0194535	0.0120000	0.1140000
I	47	0.0353404	0.0129787	0.0150000	0.0760000
MI	47	0.2340213	0.0693390	0.0650000	0.4210000
MC	47	0.1692979	0.0478112	0.0830000	0.2720000
CI	47	0.0561064	0.0242740	0.0240000	0.1440000
MCI	47	0.3544255	0.0657934	0.2160000	0.4840000

Initial Factor Method: Principal Components

Prior Communality Estimates: ONE

Eigenvalues of the Covariance Matrix:
Total = 0.01587276 Average = 0.00226754

	1	2	3	4
Eigenvalue	0.0081	0.0046	0.0021	0.0008
Difference	0.0035	0.0025	0.0013	0.0006
Proportion	0.5088	0.2872	0.1325	0.0501
Cumulative	0.5088	0.7960	0.9285	0.9787

	5	6	7
Eigenvalue	0.0002	0.0001	0.0000
Difference	0.0001	0.0001	
Proportion	0.0131	0.0082	0.0000
Cumulative	0.9918	1.0000	1.0000

3 factors will be retained by the NFACTOR criterion.

Factor Pattern

	FACTOR1	FACTOR2	FACTOR3
M	-0.67140	0.56859	0.44351
C	0.30724	0.47428	-0.23686
I	-0.21608	0.02492	-0.18149
MI	-0.78181	-0.59349	-0.18454
MC	0.57950	0.47349	-0.60049
CI	0.39917	0.39295	0.09424
MCI	0.79531	-0.50530	0.31587

Variance explained by each factor

	FACTOR1	FACTOR2	FACTOR3
Weighted	0.008076	0.004559	0.002104
Unweighted	2.[illegible]774	1.535019	0.789035

Final Communality Estimates and Variable Weights
Total Communality: Weighted = 0.014738 Unweighted = 4.654828

	M	C	I	MI
Communality	0.970776	0.375436	0.080253	0.997522
Weight	0.003314	0.000378	0.000168	0.004808

	MC	CI	MCI
Communality	0.920595	0.322632	0.987615
Weight	0.002286	0.000589	0.004329

Rotation Method: Varimax

Orthogonal Transformation Matrix

	1	2	3
1	0.54283	-0.52861	0.65262
2	0.77335	0.61765	-0.14296
3	-0.32752	0.58230	0.74408

Rotated Factor Pattern

	FACTOR1	FACTOR2	FACTOR3
M	-0.07000	0.96436	-0.18944
C	0.61113	-0.00739	-0.04354
I	-0.03858	0.02393	-0.27963
MI	-0.82293	-0.06075	-0.56269
MC	0.87741	-0.36355	-0.13631
CI	0.48971	0.08657	0.27445
MCI	-0.06252	-0.54858	0.82630

Variance explained by each factor

	FACTOR1	FACTOR2	FACTOR3
Weighted	0.005332	0.004709	0.004697
Unweighted	2.070657	1.374910	1.209262

Final Communality Estimates and Variable Weights
Total Communality: Weighted = 0.014738 Unweighted = 4.654828

	M	C	I	MI
Communality	0.970776	0.375436	0.080253	0.997522
Weight	0.003314	0.000378	0.000168	0.004808

	MC	CI	MCI
Communality	0.920595	0.322632	0.987615
Weight	0.002286	0.000589	0.004329

Scoring Coefficients Estimated by Regression

Squared Multiple Correlations of the Variables with each Factor

FACTOR1	FACTOR2	FACTOR3
1.000000	1.000000	1.000000

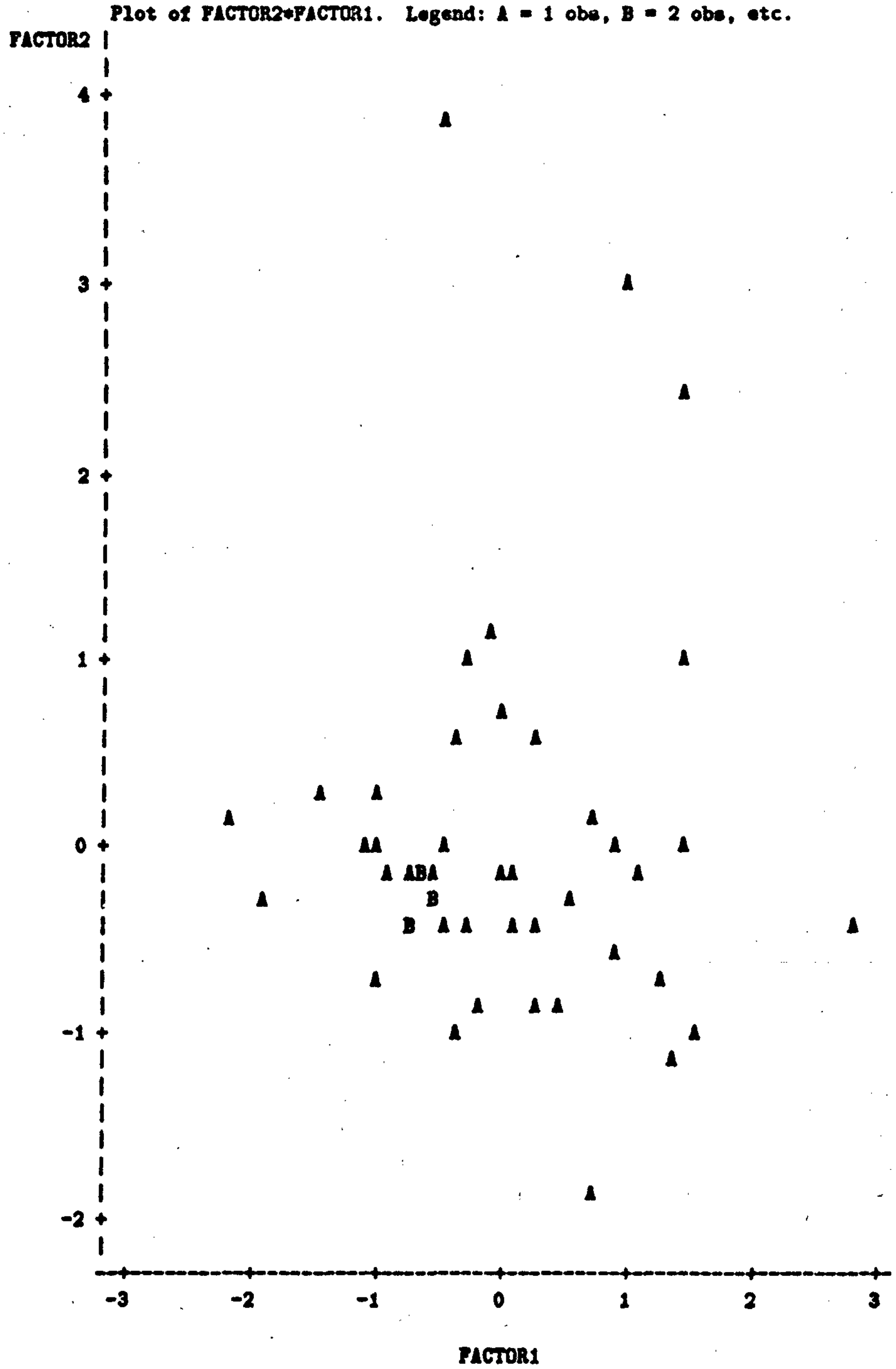
Rotation Method: Varimax

Standardized Scoring Coefficients

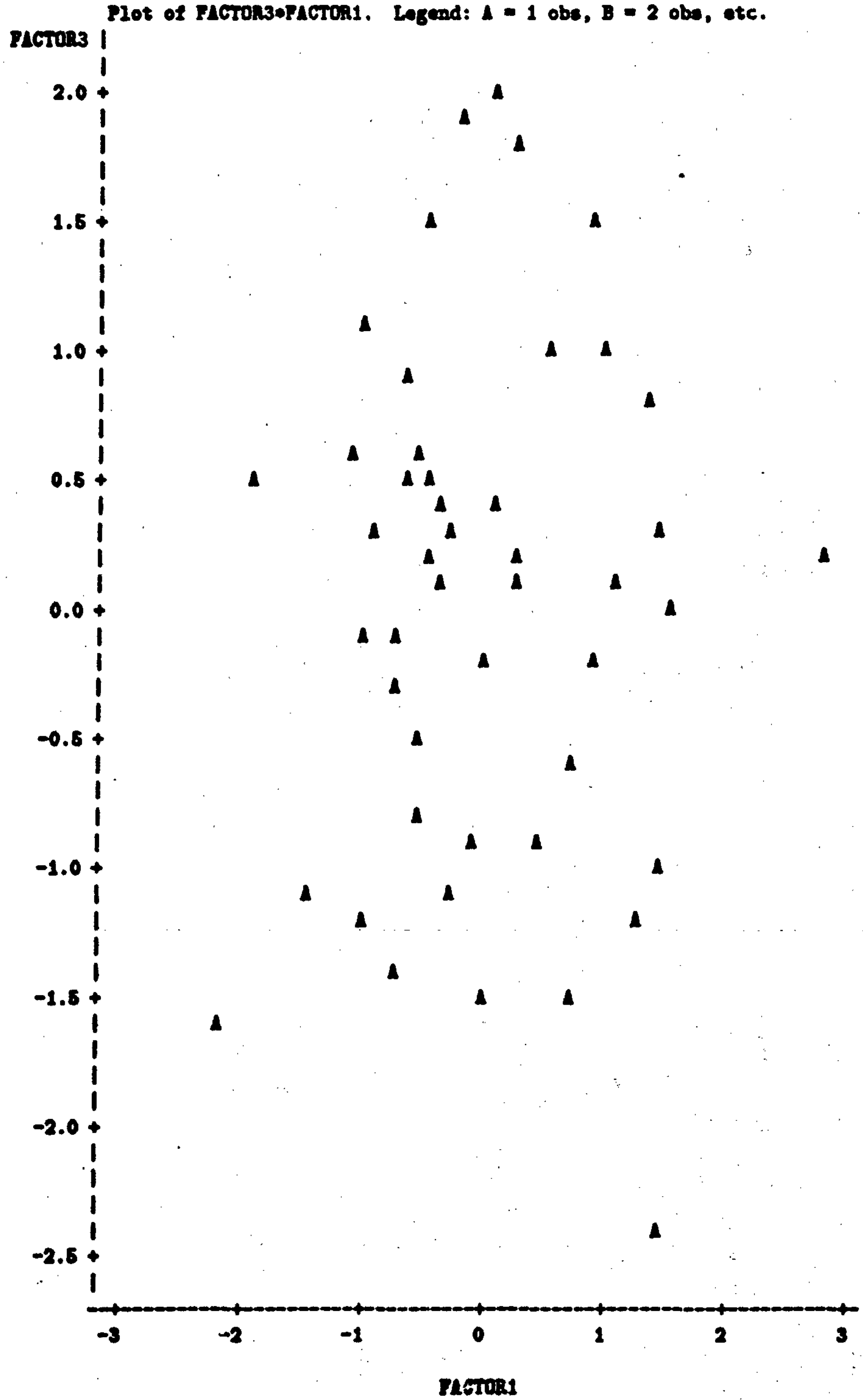
	FACTOR1	FACTOR2	FACTOR3
M	-0.05874	0.80778	0.28096
C	0.05222	-0.00810	-0.02794
I	0.00302	-0.00551	-0.01389
MI	-0.59858	-0.38613	-0.52808
MC	0.48634	-0.32001	-0.41238
CI	0.04644	0.03134	0.03139
MCI	-0.35251	-0.14324	0.83040

OBS	FACTOR1	FACTOR2	FACTOR3
1	0.30599	-0.86848	0.24602
2	0.41129	-0.87436	-0.92352
3	-0.10170	1.09098	-0.90960
4	-0.56469	-0.32118	0.56797
5	0.23733	-0.41422	0.13500
6	-1.10447	0.04215	0.56772
7	-1.94219	-0.23229	0.51099
8	-0.26817	0.96796	0.31962
9	1.34803	-1.12803	0.77735
10	-0.65264	-0.11756	0.90990
11	-0.20344	-0.90023	1.85896
12	0.92024	-0.06585	-0.22662
13	-0.36257	-0.98188	0.36834
14	-2.16416	0.16721	-1.63606
15	1.44181	0.94893	0.26029
16	1.01814	2.93499	1.00966
17	-0.56993	-0.21268	-0.75004
18	-1.00115	0.27298	1.05552
19	0.01189	-0.15656	-0.23989
20	-0.42773	-0.47789	1.53791
21	1.45958	-0.01932	-2.39613
22	0.70139	0.18718	-0.63886
23	-0.74331	-0.48370	-1.40970
24	-0.57107	-0.31938	-0.45719
25	-0.90544	-0.20446	0.27272
26	-1.02391	0.06770	-1.16823
27	-0.48460	3.88417	0.15069
28	1.09930	-0.13779	0.13130
29	-0.95876	-0.65489	-0.10876
30	1.26348	-0.74532	-1.19473
31	0.25584	0.50128	1.79467
32	-0.24719	-0.44104	-1.07830
33	0.75965	-1.80443	-1.52314
34	2.82205	-0.44477	0.16445
35	0.08196	-0.36963	1.97096
36	0.04668	-0.13165	0.41209
37	-0.34220	0.51716	0.14301
38	-0.59833	-0.17240	0.52130
39	-1.46788	0.34997	-1.06101
40	0.87045	-0.53515	1.52719
41	1.58695	-0.95398	-0.04258
42	-0.70490	-0.12445	-0.10293
43	-0.01177	0.64370	-1.48063
44	0.54622	-0.23917	0.98802
45	-0.75005	-0.47180	-0.32710
46	1.48278	2.48058	-0.99428
47	-0.49877	-0.05241	0.46764

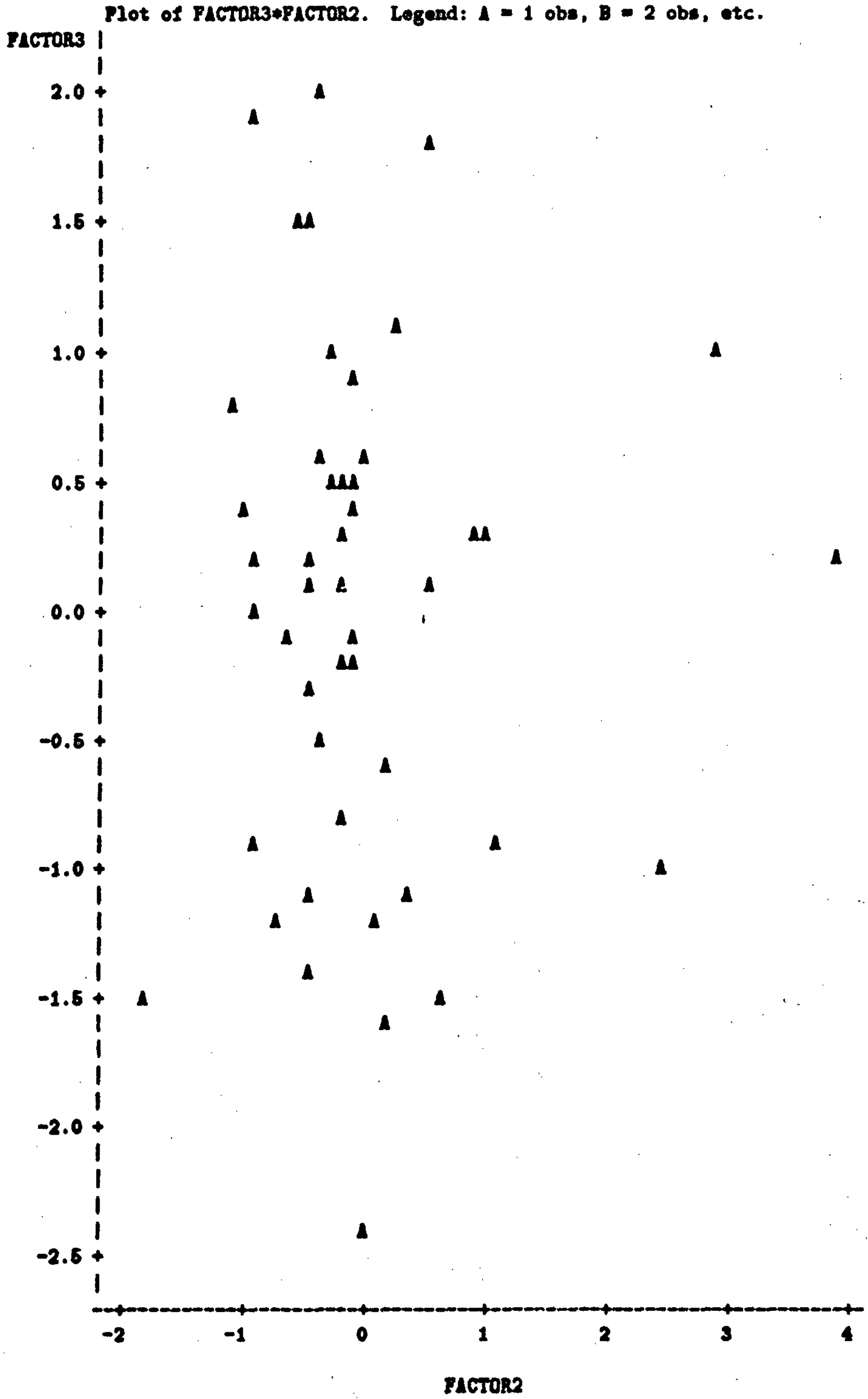
Plot of FACTOR2*FACTOR1. Legend: A = 1 obs, B = 2 obs, etc.



Plot of FACTOR3*FACTOR1. Legend: A = 1 obs, B = 2 obs, etc.



Plot of FACTOR3*FACTOR2. Legend: A = 1 obs, B = 2 obs, etc.



Complete Linkage Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	0.697871	0.638318	0.874651	0.87465
2	0.059553	0.037518	0.074639	0.94929
3	0.022035	0.007156	0.027617	0.97691
4	0.014879	0.012880	0.018648	0.99555
5	0.001999	0.000452	0.002506	0.99806
6	0.001547	0.001547	0.001939	1.00000
7	0.000000	0.000000	0.000000	1.00000
8	0.000000	0.000000	0.000000	1.00000
9	0.000000	0.000000	0.000000	1.00000
10	0.000000	0.000000	0.000000	1.00000
11	0.000000	0.000000	0.000000	1.00000
12	0.000000	0.000000	0.000000	1.00000
13	0.000000	0.000000	0.000000	1.00000
14	0.000000	0.000000	0.000000	1.00000
15	0.000000	0.000000	0.000000	1.00000
16	0.000000	0.000000	0.000000	1.00000
17	0.000000	0.000000	0.000000	1.00000
18	0.000000	0.000000	0.000000	1.00000
19	0.000000	0.000000	0.000000	1.00000
20	0.000000	0.000000	0.000000	1.00000
21	0.000000	0.000000	0.000000	1.00000
22	0.000000	0.000000	0.000000	1.00000
23	0.000000	0.000000	0.000000	1.00000
24	0.000000	0.000000	0.000000	1.00000
25	0.000000	0.000000	0.000000	1.00000
26	0.000000	0.000000	0.000000	1.00000
27	-.000000	0.000000	-.000000	1.00000
28	-.000000	0.000000	-.000000	1.00000
29	-.000000	0.000000	-.000000	1.00000
30	-.000000	0.000000	-.000000	1.00000
31	-.000000	0.000000	-.000000	1.00000
32	-.000000	0.000000	-.000000	1.00000
33	-.000000	0.000000	-.000000	1.00000
34	-.000000	0.000000	-.000000	1.00000
35	-.000000	0.000000	-.000000	1.00000
36	-.000000	0.000000	-.000000	1.00000
37	-.000000	0.000000	-.000000	1.00000
38	-.000000	0.000000	-.000000	1.00000
39	-.000000	0.000000	-.000000	1.00000
40	-.000000	0.000000	-.000000	1.00000
41	-.000000	0.000000	-.000000	1.00000
42	-.000000	0.000000	-.000000	1.00000
43	-.000000	0.000000	-.000000	1.00000
44	-.000000	0.000000	-.000000	1.00000
45	-.000000	0.000000	-.000000	1.00000
46	-.000000	0.000000	-.000000	1.00000
47	-.000000	.	-.000000	1.00000

Root-Mean-Square Total-Sample Standard Deviation = 0.130293
Mean Distance Between Observations = 1.107356

Complete Linkage Cluster Analysis

Number of Clusters	Clusters Joined		Frequency of New Cluster	Normalized Maximum Distance	Tie
6	OB2	OB3	2	0.154549	
5	CL6	OB6	3	0.214065	
4	OB1	OB5	2	0.604346	
3	CL4	OB4	3	0.866503	
2	CL3	CL5	6	1.364139	
1	CL2	OB7	7	2.062808	

```
                          The SAS System                        12
                                  10:54 Thursday, December 17, 1992
              Complete Linkage Cluster Analysis
                   Name of Observation or Cluster
        OB     OB     OB     OB     OB     OB     OB
        1      5      4      2      3      6      7
  2.25 +
       |
       |
       |
       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
     2 +XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
M      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
a      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
x      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
i      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
m 1.75 +XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
u      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
m      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
D      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
i  1.5 +XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
s      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
t      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
a      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .
n      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
c 1.25 +XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
e      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
       |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
B      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
e      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
t    1 +XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
w      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
e      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
e      |XXXXXXXXXXXXXXX      XXXXXXXXXXXXXXX      .
n      |XXXXXXXX      .      XXXXXXXXXXXXXXX      .
  0.75 +XXXXXXXX      .      XXXXXXXXXXXXXXX      .
C      |XXXXXXXX      .      XXXXXXXXXXXXXXX      .
l      |XXXXXXXX      .      XXXXXXXXXXXXXXX      .
u      |XXXXXXXX      .      XXXXXXXXXXXXXXX      .
s      |.      .      .      XXXXXXXXXXXXXXX      .
t  0.5 +.      .      .      XXXXXXXXXXXXXXX      .
e      |.      .      .      XXXXXXXXXXXXXXX      .
r      |.      .      .      XXXXXXXXXXXXXXX      .
s      |.      .      .      XXXXXXXXXXXXXXX      .
       |.      .      .      XXXXXXXXXXXXXXX      .
  0.25 +.      .      .      XXXXXXXXXXXXXXX      .
       |.      .      .      XXXXXXXXXXXXXXX      .
       |.      .      .      XXXXXXXX      .      .
       |.      .      .      .      .      .      .
       |.      .      .      .      .      .      .
     0 +.      .      .      .      .      .      .
```

Average Linkage Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	0.697871	0.638318	0.874651	0.87465
2	0.059553	0.037518	0.074639	0.94929
3	0.022035	0.007156	0.027617	0.97691
4	0.014879	0.012880	0.018648	0.99555
5	0.001999	0.000452	0.002506	0.99806
6	0.001547	0.001547	0.001939	1.00000
7	0.000000	0.000000	0.000000	1.00000
8	0.000000	0.000000	0.000000	1.00000
9	0.000000	0.000000	0.000000	1.00000
10	0.000000	0.000000	0.000000	1.00000
11	0.000000	0.000000	0.000000	1.00000
12	0.000000	0.000000	0.000000	1.00000
13	0.000000	0.000000	0.000000	1.00000
14	0.000000	0.000000	0.000000	1.00000
15	0.000000	0.000000	0.000000	1.00000
16	0.000000	0.000000	0.000000	1.00000
17	0.000000	0.000000	0.000000	1.00000
18	0.000000	0.000000	0.000000	1.00000
19	0.000000	0.000000	0.000000	1.00000
20	0.000000	0.000000	0.000000	1.00000
21	0.000000	0.000000	0.000000	1.00000
22	0.000000	0.000000	0.000000	1.00000
23	0.000000	0.000000	0.000000	1.00000
24	0.000000	0.000000	0.000000	1.00000
25	0.000000	0.000000	0.000000	1.00000
26	0.000000	0.000000	0.000000	1.00000
27	-.000000	0.000000	-.000000	1.00000
28	-.000000	0.000000	-.000000	1.00000
29	-.000000	0.000000	-.000000	1.00000
30	-.000000	0.000000	-.000000	1.00000
31	-.000000	0.000000	-.000000	1.00000
32	-.000000	0.000000	-.000000	1.00000
33	-.000000	0.000000	-.000000	1.00000
34	-.000000	0.000000	-.000000	1.00000
35	-.000000	0.000000	-.000000	1.00000
36	-.000000	0.000000	-.000000	1.00000
37	-.000000	0.000000	-.000000	1.00000
38	-.000000	0.000000	-.000000	1.00000
39	-.000000	0.000000	-.000000	1.00000
40	-.000000	0.000000	-.000000	1.00000
41	-.000000	0.000000	-.000000	1.00000
42	-.000000	0.000000	-.000000	1.00000
43	-.000000	0.000000	-.000000	1.00000
44	-.000000	0.000000	-.000000	1.00000
45	-.000000	0.000000	-.000000	1.00000
46	-.000000	0.000000	-.000000	1.00000
47	-.000000	.	-.000000	1.00000

Root-Mean-Square Total-Sample Standard Deviation = 0.130293
Root-Mean-Square Distance Between Observations = 1.263238

Average Linkage Cluster Analysis

Number of Clusters	Clusters Joined		Frequency of New Cluster	Pseudo F	Pseudo t**2	Normalized RMS Distance	Tie
6	OB2	OB3	2	65.18	.	0.135478	
5	CL6	OB6	3	51.57	2.14	0.184487	
4	OB1	OB5	2	16.74	.	0.529770	
3	CL4	CL5	5	7.99	7.65	0.662877	
2	OB4	OB7	2	10.07	.	0.888342	
1	CL3	CL2	7	.	10.07	1.311767	

Average Linkage Cluster Analysis

```
                   Name of Observation or Cluster
          OB     OB     OB     OB     OB     OB     OB
          1      5      2      3      6      4      7
    1.4 +
        |
        |
        |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
        |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
        |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
        |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
    1.2 +XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
A       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
v       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
e       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
r       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
a       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
g       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
e     1 +XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
        |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
D       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
i       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
s       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      XXXXXXX
t       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
a       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
n   0.8 +XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
c       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
e       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
        |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
B       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
e       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXX      .      .
t       |XXXXXXXX      XXXXXXXXXXXXXXX      .      .
w   0.6 +XXXXXXXX      XXXXXXXXXXXXXXX      .      .
e       |XXXXXXXX      XXXXXXXXXXXXXXX      .      .
e       |XXXXXXXX      XXXXXXXXXXXXXXX      .      .
n       |.      .      XXXXXXXXXXXXXXX      .      .
        |.      .      XXXXXXXXXXXXXXX      .      .
C       |.      .      XXXXXXXXXXXXXXX      .      .
l       |.      .      XXXXXXXXXXXXXXX      .      .
u   0.4 +.      .      XXXXXXXXXXXXXXX      .      .
s       |.      .      XXXXXXXXXXXXXXX      .      .
t       |.      .      XXXXXXXXXXXXXXX      .      .
e       |.      .      XXXXXXXXXXXXXXX      .      .
r       |.      .      XXXXXXXXXXXXXXX      .      .
s       |.      .      XXXXXXXXXXXXXXX      .      .
        |.      .      XXXXXXXXXXXXXXX      .      .
    0.2 +.      .      XXXXXXXXXXXXXXX      .      .
        |.      .      XXXXXXXXXXXXXXX      .      .
        |.      .      XXXXXXXX      .      .      .
        |.      .      .      .      .      .      .
        |.      .      .      .      .      .      .
        |.      .      .      .      .      .      .
        |.      .      .      .      .      .      .
      0 +.      .      .      .      .      .      .
```

Ward's Minimum Variance Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	0.697871	0.638318	0.874651	0.87465
2	0.059553	0.037518	0.074639	0.94929
3	0.022035	0.007156	0.027617	0.97691
4	0.014879	0.012880	0.018648	0.99555
5	0.001999	0.000452	0.002506	0.99806
6	0.001547	0.001547	0.001939	1.00000
7	0.000000	0.000000	0.000000	1.00000
8	0.000000	0.000000	0.000000	1.00000
9	0.000000	0.000000	0.000000	1.00000
10	0.000000	0.000000	0.000000	1.00000
11	0.000000	0.000000	0.000000	1.00000
12	0.000000	0.000000	0.000000	1.00000
13	0.000000	0.000000	0.000000	1.00000
14	0.000000	0.000000	0.000000	1.00000
15	0.000000	0.000000	0.000000	1.00000
16	0.000000	0.000000	0.000000	1.00000
17	0.000000	0.000000	0.000000	1.00000
18	0.000000	0.000000	0.000000	1.00000
19	0.000000	0.000000	0.000000	1.00000
20	0.000000	0.000000	0.000000	1.00000
21	0.000000	0.000000	0.000000	1.00000
22	0.000000	0.000000	0.000000	1.00000
23	0.000000	0.000000	0.000000	1.00000
24	0.000000	0.000000	0.000000	1.00000
25	0.000000	0.000000	0.000000	1.00000
26	0.000000	0.000000	0.000000	1.00000
27	-.000000	0.000000	-.000000	1.00000
28	-.000000	0.000000	-.000000	1.00000
29	-.000000	0.000000	-.000000	1.00000
30	-.000000	0.000000	-.000000	1.00000
31	-.000000	0.000000	-.000000	1.00000
32	-.000000	0.000000	-.000000	1.00000
33	-.000000	0.000000	-.000000	1.00000
34	-.000000	0.000000	-.000000	1.00000
35	-.000000	0.000000	-.000000	1.00000
36	-.000000	0.000000	-.000000	1.00000
37	-.000000	0.000000	-.000000	1.00000
38	-.000000	0.000000	-.000000	1.00000
39	-.000000	0.000000	-.000000	1.00000
40	-.000000	0.000000	-.000000	1.00000
41	-.000000	0.000000	-.000000	1.00000
42	-.000000	0.000000	-.000000	1.00000
43	-.000000	0.000000	-.000000	1.00000
44	-.000000	0.000000	-.000000	1.00000
45	-.000000	0.000000	-.000000	1.00000
46	-.000000	0.000000	-.000000	1.00000
47	-.000000	.	-.000000	1.00000

Root-Mean-Square Total-Sample Standard Deviation = 0.130293
Root-Mean-Square Distance Between Observations = 1.263238

Ward's Minimum Variance Cluster Analysis

NCL	Clusters Joined		FREQ	SPRSQ	RSQ	Pseudo F	Pseudo t**2	Tie
6	OB2	OB3	2	0.003059	0.996941	65.2	.	
5	CL6	OB6	3	0.006544	0.990397	51.6	2.1	
4	OB1	OB5	2	0.046776	0.943621	16.7	.	
3	CL4	OB4	3	0.097934	0.845687	11.0	2.1	
2	CL3	CL5	6	0.284481	0.561206	6.4	7.4	
1	CL2	OB7	7	0.561206	0.000000	.	6.4	

Ward's Minimum Variance Cluster Analysis

```
            Name of Observation or Cluster
       OB   OB   OB   OB   OB   OB   OB
       1    5    4    2    3    6    7
  0.6 +
      |
      |
      |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
  0.5 +XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
S     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
e     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
m     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
i 0.4 +XXXXXXXXXXXXXXXXXXXXXXXXXX    .
-     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
P     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
a     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
r     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
t     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
i     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
a     |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
l 0.3 +XXXXXXXXXXXXXXXXXXXXXXXXXX    .
      |XXXXXXXXXXXXXXXXXXXXXXXXXX    .
R     |XXXXXXXXXXX    XXXXXXXXXXX    .
-     |XXXXXXXXXXX    XXXXXXXXXXX    .
S     |XXXXXXXXXXX    XXXXXXXXXXX    .
q     |XXXXXXXXXXX    XXXXXXXXXXX    .
u     |XXXXXXXXXXX    XXXXXXXXXXX    .
a     |XXXXXXXXXXX    XXXXXXXXXXX    .
r 0.2 +XXXXXXXXXXX    XXXXXXXXXXX    .
e     |XXXXXXXXXXX    XXXXXXXXXXX    .
d     |XXXXXXXXXXX    XXXXXXXXXXX    .
      |XXXXXXXXXXX    XXXXXXXXXXX    .
      |XXXXXXXXXXX    XXXXXXXXXXX    .
      |XXXXXXXXXXX    XXXXXXXXXXX    .
      |XXXXXXXXXXX    XXXXXXXXXXX    .
      |XXXXXXXXXXX    XXXXXXXXXXX    .
  0.1 +XXXXXXXXXXX    XXXXXXXXXXX    .
      |XXXXXX    .    XXXXXXXXXXX    .
      |XXXXXX    .    XXXXXXXXXXX    .
      |XXXXXX    .    XXXXXXXXXXX    .
      |XXXXXX    .    XXXXXXXXXXX    .
      |.    .    .    XXXXXXXXXXX    .
      |.    .    .    XXXXXXXXXXX    .
      |.    .    .    XXXXXXXXXXX    .
    0 +.    .    .    XXXXXX    .    .
```

OBS	NP	NT	CP	CT	IP	IT	NIP	NIT	NCP	NCT	CIP	CIT	NCIP	NCIT
1	1	33	1	37	0	57	0	60	1	31	1	59	1	42
2	1	60	0	60	1	40	1	35	1	38	0	53	1	47
3	1	27	0	60	0	60	1	18	1	40	1	60	1	57
4	0	60	1	60	1	60	1	60	1	30	0	60	1	60
5	0	60	0	60	1	37	1	60	1	30	1	60	1	34
6	1	60	0	60	1	60	1	40	1	16	0	60	1	32
7	1	40	0	60	0	52	1	26	1	14	0	22	0	60
8	0	60	1	19	1	58	1	30	1	20	0	60	1	60
9	0	60	0	60	0	60	0	60	0	60	0	60	0	60
10	1	43	0	60	0	60	1	39	1	60	0	60	1	50
11	1	35	1	34	0	31	1	17	1	33	1	27	1	57
12	1	30	1	43	1	33	1	31	1	27	[illegible]	33	1	28
13	0	60	0	60	1	60	0	60	1	28	0	60	1	27
14	0	60	0	60	0	18	1	22	1	57	0	60	0	60
15	1	35	1	38	0	60	1	34	1	31	1	60	1	23
16		60	0	60	1	33	0	28	1	18	0	60	0	39
17	1	30	0	60	0	30	1	30	1	17	0	60	1	28
18	1	38	1	33	0	60	1	36	1	56	1	42	1	26
19	1	24	1	18	1	60	1	33	1	20	1	28	1	30
20	1	26	1	56	1	28	0	60	1	40	1	60	1	14
21	1	30	1	30	0	60	1	31	1	16	1	60	1	30
22	1	30	1	50	1	47	1	46	1	29	1	42	1	18
23	1	28	1	49	0	19	0	14	1	15	1	20	1	26
24	1	24	1	39	1	31	0	19	1	26	1	30	1	18
25	1	27	1	57	0	60	0	60	1	31	0	60	1	39
26	1	47	0	60	1	60	1	52	1	39	1	60	1	60
27	1	25	0	60	1	60	1	17	1	24	0	60	1	30
28	1	30	0	60	0	39	1	29	1	12	1	54	1	24
29	1	33	0	60	1	60	1	11	0	19	1	23	1	20
30	1	37	1	39	1	35	1	34	1	30	0	60	0	60
31	1	21	0	60	0	46	1	24	0	60	1	39	1	47
32	1	23	1	38	1	40	1	17	1	24	1	26	1	19
33	1	25	1	36	0	58	1	32	1	16	0	60	1	27
34	1	22	1	20	1	60	0	38	1	29	1	56	1	22
35	0	60	1	60	0	60	1	17	1	28	0	60	1	18
36	1	25	1	25	1	60	1	17	1	25	1	30	1	20
37	1	8	0	60	1	40	1	26	1	26	0	60	1	16
38	1	27	1	28	1	60	1	10	1	27	1	12	1	19
39	1	19	1	50	0	60	1	41	1	47	1	17	1	28
40	1	42	0	20	1	37	1	19	1	26	0	9	1	20
41	1	42	1	20	0	60	1	35	1	19	1	60	1	29
42	1	34	0	60	0	60	1	51	1	29	0	60	1	21
43	1	28	1	41	0	37	1	34	1	14	0	9	0	9
44	0	54	0	60	0	60	1	28	1	30	1	23	1	44
45	1	25	0	60	1	38	1	33	1	44	1	60	1	26
46	1	27	1	37	0	31	1	12	1	38	1	19	1	53
47	1	21	1	15	1	20	1	27	1	26	1	28	1	9

Variable	N	Mean	Std Dev	Minimum	Maximum
MP	47	0.8085106	0.3977271	0	1.0000000
MT	47	36.4893617	14.5600292	8.0000000	60.0000000
CP	47	0.5531915	0.5025375	0	1.0000000
CT	47	46.8085106	15.2581244	15.0000000	60.0000000
IP	47	0.5106383	0.5052912	0	1.0000000
IT	47	47.9787234	13.9416320	18.0000000	60.0000000
MIP	47	0.8085106	0.3977271	0	1.0000000
MIT	47	33.4255319	14.9981806	10.0000000	60.0000000
MCP	47	0.9361702	0.2470922	0	1.0000000
MCT	47	30.1063830	12.8574696	12.0000000	60.0000000
CIP	47	0.5744681	0.4997687	0	1.0000000
CIT	47	45.5531915	18.0708852	9.0000000	60.0000000
MCIP	47	0.8723404	0.3373181	0	1.0000000
MCIT	47	33.7446809	15.9788829	9.0000000	60.0000000

OBS	MRP	CRP	IRP	MIRP	MCRP	CIRP	MCIRP
1	33	37	40	35	31	59	42
2	60	60	60	18	38	60	47
3	27	19	37	60	40	60	57
4	60	34	60	60	30	27	60
5	40	43	58	40	30	33	34
6	43	38	33	26	16	60	32
7	35	33	60	30	14	42	60
8	30	18	33	39	20	28	50
9	35	56	60	17	60	60	57
10	30	38	28	31	33	60	28
11	38	50	47	22	27	42	27
12	24	49	31	34	28	20	23
13	26	39	60	30	57	30	28
14	30	57	60	36	31	60	26
15	30	39	60	33	18	54	30
16	28	38	35	31	17	23	14
17	24	36	40	46	56	39	30
18	27	20	60	52	20	26	18
19	37	60	60	17	40	56	26
20	25	25	40	29	16	30	18
21	30	28	60	11	29	12	39
22	33	50	37	34	15	17	60
23	37	20	38	44	26	60	30
24	21	41	20	17	31	23	24
25	23	37	.	32	39	60	20
26	25	15	.	17	24	19	47
27	22	.	.	17	12	28	19
28	25	.	.	26	30	.	27
29	8	.	.	10	24	.	22
30	27	.	.	41	16	.	18
31	19	.	.	19	29	.	20
32	42	.	.	35	28	.	16
33	42	.	.	51	25	.	19
34	34	.	.	34	26	.	28
35	28	.	.	28	27	.	20
36	25	.	.	33	47	.	29
37	27	.	.	12	26	.	21
38	21	.	.	27	19	.	44
39	.	.	.	.	29	.	26
40	.	.	.	.	14	.	53
41	.	.	.	.	30	.	9
42	.	.	.	.	44	.	.
43	.	.	.	.	38	.	.
44	.	.	.	.	26	.	.

Variable	N	Mean	Std Dev	Minimum	Maximum
MRP	38	31.0789474	10.2808707	8.0000000	60.0000000
CRP	26	37.6923077	13.1902062	15.0000000	60.0000000
IRP	24	46.5416667	13.3773955	20.0000000	60.0000000
MIRP	38	30.8947368	12.6401109	10.0000000	60.0000000
MCRP	44	29.0000000	11.4526325	12.0000000	60.0000000
CIRP	27	40.2962963	17.1572250	12.0000000	60.0000000
MCIRP	41	31.6585366	14.2909233	9.0000000	60.0000000

OBS	MWP	CWP	IWP	MIWP	MCWP	CIWP	MCIWP
1	60	60	57	60	60	53	60
2	60	60	60	60	19	60	50
3	60	60	52	60	60	60	60
4	60	60	60	26	.	22	39
5	60	60	60	60	.	60	60
6	60	60	31	14	.	60	9
7	60	60	18	19	.	60	.
8	60	60	60	60	.	60	.
9	54	60	30	38	.	60	.
10	.	60	60	.	.	60	.
11	.	60	60	.	.	60	.
12	.	60	19	.	.	60	.
13	.	60	60	.	.	60	.
14	.	60	39	.	.	60	.
15	.	60	46	.	.	60	.
16	.	60	58	.	.	60	.
17	.	60	60	.	.	60	.
18	.	20	60	.	.	9	.
19	.	60	60	.	.	60	.
20	.	60	60	.	.	9	.
21	.	60	37	.	.	.	.
22	.	.	60	.	.	.	.
23	.	.	31	.	.	.	.

Variable	N	Mean	Std Dev	Minimum	Maximum
MWP	9	59.3333333	2.0000000	54.0000000	60.0000000
CWP	21	58.0952381	8.7287156	20.0000000	60.0000000
IWP	23	49.4782609	14.6531708	18.0000000	60.0000000
MIWP	9	44.1111111	19.8899751	14.0000000	60.0000000
MCWP	3	46.3333333	23.6713610	19.0000000	60.0000000
CIWP	20	52.6500000	17.1901900	9.0000000	60.0000000
MCIWP	6	48.0000000	20.8710326	9.0000000	60.0000000

OBS	MNA	MNT	CNA	CET	INA	IET	MINA	MINT	MCNA	MCET	CINA	CIET	MCINA	MCINT
1	1	33	1	37	0	57	1	35	1	31	1	59	1	42
2	1	27	1	19	1	40	1	18	1	38	0	53	1	47
3	1	40	1	34	1	37	1	40	1	40	0	22	1	57
4	1	43	1	43	0	52	1	26	1	30	1	27	1	34
5	1	35	1	38	1	58	1	30	1	30	1	33	1	32
6	1	30	1	33	0	31	1	39	1	16	1	43	1	50
7	1	35	1	18	1	33	1	17	1	14	1	28	1	57
8	1	30	1	56	0	18	1	31	1	20	1	42	1	28
9	1	38	1	38	1	33	1	22	1	33	1	20	1	27
10	1	24	1	50	0	30	1	34	1	27	1	30	1	23
11	1	26	1	49	1	28	0	26	1	28	1	54	0	39
12	1	30	1	39	1	47	1	30	1	57	1	23	1	28
13	1	30	1	57	0	19	1	36	1	31	1	39	1	26
14	1	28	1	39	1	31	1	33	1	18	1	26	1	30
15	1	24	1	38	0	39	1	31	1	17	1	56	1	14
16	1	27	1	36	1	35	1	46	1	56	1	30	1	30
17	1	47	1	20	0	46	0	14	1	20	1	12	1	18
18	1	25	1	25	1	40	0	19	1	40	1	17	1	26
19	1	30	1	28	0	58	1	52	1	16	0	9	1	18
20	1	33	1	50	1	40	1	17	1	29	0	9	1	39
21	1	37	0	20	1	37	1	29	1	15	1	23	1	30
22	1	21	1	20	0	37	1	11	1	26	1	19	1	24
23	1	23	1	41	1	38	1	34	1	31	1	28	1	20
24	1	25	1	37	0	31	1	44	1	39	.	.	1	47
25	1	22	1	15	1	20	1	17	1	24	.	.	1	19
26	1	25	.	.	.	.	1	32	1	12	.	.	1	27
27	1	8	.	.	.	.	0	38	0	19	.	.	1	22
28	1	27	.	.	.	.	1	17	1	30	.	.	1	18
29	1	19	.	.	.	.	1	17	1	24	.	.	1	20
30	1	42	.	.	.	.	1	26	1	16	.	.	1	16
31	1	42	.	.	.	.	1	10	1	29	.	.	1	19
32	1	34	.	.	.	.	1	41	1	28	.	.	1	28
33	1	28	.	.	.	.	1	19	1	25	.	.	1	20
34	0	54	.	.	.	.	1	35	1	26	.	.	1	29
35	1	25	.	.	.	.	1	51	1	27	.	.	1	21
36	1	27	.	.	.	.	1	34	1	47	.	.	0	9
37	1	21	.	.	.	.	1	28	1	26	.	.	1	44
38	.	.	.	.	.	.	1	33	1	19	.	.	1	26
39	.	.	.	.	.	.	1	12	1	29	.	.	1	53
40	.	.	.	.	.	.	1	27	1	14	.	.	1	9
41	.	.	.	.	.	.	.	.	1	30	.	.	.	.
42	.	.	.	.	.		.	.	1	44	.	.	.	.
43	.	.	.	.	.	.	.	.	1	38	.	.	.	.
44	.	.	.	.	.	.	.	.	1	26	.	.	.	.

Variable	N	Mean	Std Dev	Minimum	Maximum
MNA	37	0.9729730	0.1643990	0	1.0000000
MNT	37	30.1351351	8.7087509	8.0000000	54.0000000
CNA	25	0.9600000	0.2000000	0	1.0000000
CNT	25	35.2000000	12.0968315	15.0000000	57.0000000
INA	25	0.5600000	0.5066228	0	1.0000000
INT	25	37.4000000	11.1130554	18.0000000	58.0000000
MINA	40	0.9000000	0.3038218	0	1.0000000
MINT	40	28.7750000	10.7881404	10.0000000	52.0000000
MCNA	44	0.9772727	0.1507557	0	1.0000000
MCNT	44	28.0681818	10.5000503	12.0000000	57.0000000
CINA	23	0.8260870	0.3875534	0	1.0000000
CINT	23	30.4782609	14.7490202	9.0000000	59.0000000
MCINA	40	0.9500000	0.2207214	0	1.0000000
MCINT	40	29.1500000	12.4808571	9.0000000	57.0000000

The SAS System 27
10:54 Thursday, December 17, 1992

OBS	MOA	COA	IOA	MIOA	MCOA	CIOA	MCIOA
1	1	0	0	0	0	1	1
2	0	0	1	1	1	0	0
3	0	1	1	1	0	1	1
4	1	0	0	0	.	0	0
5	0	0	0	0	.	0	0
6	0	0	1	0	.	0	1
7	0	0	0	0	.	0	0
8	0	0	0	.	.	0	.
9	0	0	1	.	.	0	.
10	0	0	0	.	.	1	.
11	.	0	0	.	.	0	.
12	.	0	1	.	.	0	.
13	.	0	1	.	.	1	.
14	.	0	1	.	.	1	.
15	.	0	1	.	.	0	.
16	.	0	0	.	.	1	.
17	.	0	1	.	.	0	.
18	.	1	1	.	.	0	.
19	.	0	0	.	.	0	.
20	.	0	0	.	.	0	.
21	.	0	0	.	.	0	.
22	.	0	0	.	.	1	.
23	.	.	.	.	.	0	.
24	.	.	.	.	.	1	.

Variable	N	Mean	Std Dev	Minimum	Maximum
NOA	10	0.2000000	0.4216370	0	1.0000000
COA	22	0.0909091	0.2942449	0	1.0000000
IOA	22	0.4545455	0.5096472	0	1.0000000
NIOA	7	0.2857143	0.4879500	0	1.0000000
NCOA	3	0.3333333	0.5773503	0	1.0000000
CIOA	24	0.3333333	0.4815434	0	1.0000000
NCIOA	7	0.4285714	0.5345225	0	1.0000000

Complete Linkage Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	4024.21	2288.74	0.383053	0.38305
2	1735.47	47.18	0.165195	0.54825
3	1688.29	297.21	0.160704	0.70895
4	1391.08	360.47	0.132413	0.84137
5	1030.62	394.68	0.098101	0.93947
6	635.94	635.94	0.060533	1.00000
7	0.00	0.00	0.000000	1.00000
8	0.00	0.00	0.000000	1.00000
9	0.00	0.00	0.000000	1.00000
10	0.00	0.00	0.000000	1.00000
11	0.00	0.00	0.000000	1.00000
12	0.00	0.00	0.000000	1.00000
13	0.00	0.00	0.000000	1.00000
14	0.00	0.00	0.000000	1.00000
15	0.00	0.00	0.000000	1.00000
16	0.00	0.00	0.000000	1.00000
17	0.00	0.00	0.000000	1.00000
18	0.00	0.00	0.000000	1.00000
19	0.00	0.00	0.000000	1.00000
20	0.00	0.00	0.000000	1.00000
21	0.00	0.00	0.000000	1.00000
22	0.00	0.00	0.000000	1.00000
23	0.00	0.00	0.000000	1.00000
24	0.00	0.00	0.000000	1.00000
25	0.00	0.00	0.000000	1.00000
26	0.00	0.00	0.000000	1.00000
27	0.00	0.00	0.000000	1.00000
28	0.00	0.00	0.000000	1.00000
29	0.00	0.00	0.000000	1.00000
30	-0.00	0.00	-.000000	1.00000
31	-0.00	0.00	-.000000	1.00000
32	-0.00	0.00	-.000000	1.00000
33	-0.00	0.00	-.000000	1.00000
34	-0.00	0.00	-.000000	1.00000
35	-0.00	0.00	-.000000	1.00000
36	-0.00	0.00	-.000000	1.00000
37	-0.00	0.00	-.000000	1.00000
38	-0.00	0.00	-.000000	1.00000
39	-0.00	0.00	-.000000	1.00000
40	-0.00	0.00	-.000000	1.00000
41	-0.00	0.00	-.000000	1.00000
42	-0.00	0.00	-.000000	1.00000
43	-0.00	0.00	-.000000	1.00000
44	-0.00	0.00	-.000000	1.00000
45	-0.00	0.00	-.000000	1.00000
46	-0.00	0.00	-.000000	1.00000
47	-0.00	.	-.000000	1.00000

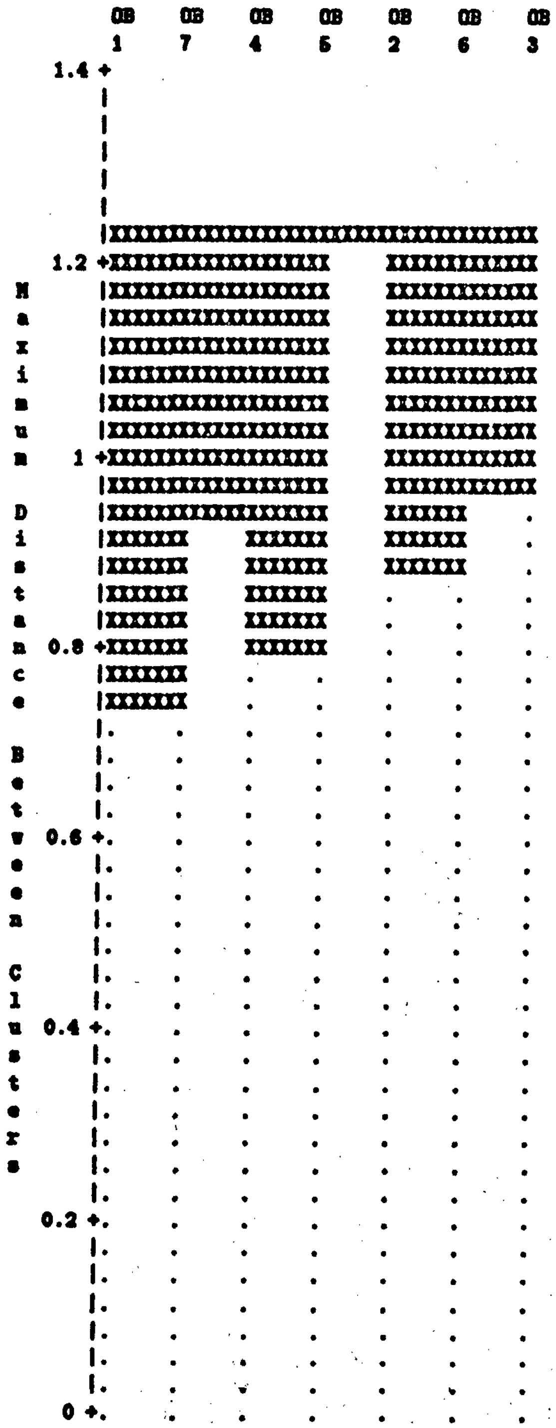
Root-Mean-Square Total-Sample Standard Deviation = 14.95071
Mean Distance Between Observations = 143.8465

Complete Linkage Cluster Analysis

Number of Clusters	Clusters Joined		Frequency of New Cluster	Normalized Maximum Distance	Tie
6	OB1	OB7	2	0.749877	
5	OB4	OB5	2	0.801696	
4	OB2	OB6	2	0.887343	
3	CL6	CL5	4	0.946333	
2	CL4	OB3	3	0.974192	
1	CL3	CL2	7	1.224518	

Complete Linkage Cluster Analysis

Name of Observation or Cluster



Average Linkage Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	4024.21	2288.74	0.383053	0.38305
2	1735.47	47.18	0.165195	0.54825
3	1688.29	297.21	0.160704	0.70895
4	1391.08	360.47	0.132413	0.84137
5	1030.62	394.68	0.098101	0.93947
6	635.94	635.94	0.060533	1.00000
7	0.00	0.00	0.000000	1.00000
8	0.00	0.00	0.000000	1.00000
9	0.00	0.00	0.000000	1.00000
10	0.00	0.00	0.000000	1.00000
11	0.00	0.00	0.000000	1.00000
12	0.00	0.00	0.000000	1.00000
13	0.00	0.00	0.000000	1.00000
14	0.00	0.00	0.000000	1.00000
15	0.00	0.00	0.000000	1.00000
16	0.00	0.00	0.000000	1.00000
17	0.00	0.00	0.000000	1.00000
18	0.00	0.00	0.000000	1.00000
19	0.00	0.00	0.000000	1.00000
20	0.00	0.00	0.000000	1.00000
21	0.00	0.00	0.000000	1.00000
22	0.00	0.00	0.000000	1.00000
23	0.00	0.00	0.000000	1.00000
24	0.00	0.00	0.000000	1.00000
25	0.00	0.00	0.000000	1.00000
26	0.00	0.00	0.000000	1.00000
27	0.00	0.00	0.000000	1.00000
28	0.00	0.00	0.000000	1.00000
29	0.00	0.00	0.000000	1.00000
30	-0.00	0.00	-.000000	1.00000
31	-0.00	0.00	-.000000	1.00000
32	-0.00	0.00	-.000000	1.00000
33	-0.00	0.00	-.000000	1.00000
34	-0.00	0.00	-.000000	1.00000
35	-0.00	0.00	-.000000	1.00000
36	-0.00	0.00	-.000000	1.00000
37	-0.00	0.00	-.000000	1.00000
38	-0.00	0.00	-.000000	1.00000
39	-0.00	0.00	-.000000	1.00000
40	-0.00	0.00	-.000000	1.00000
41	-0.00	0.00	-.000000	1.00000
42	-0.00	0.00	-.000000	1.00000
43	-0.00	0.00	-.000000	1.00000
44	-0.00	0.00	-.000000	1.00000
45	-0.00	0.00	-.000000	1.00000
46	-0.00	0.00	-.000000	1.00000
47	-0.00	.	-.000000	1.00000

Root-Mean-Square Total-Sample Standard Deviation = 14.95071
Root-Mean-Square Distance Between Observations = 144.9525

Average Linkage Cluster Analysis

Number of Clusters	Clusters Joined		Frequency of New Cluster	Pseudo F	Pseudo t**2	Normalized RMS Distance	Tie
6	OB1	OB7	2	1.97	.	0.743121	
5	OB4	OB5	2	2.04	.	0.794472	
4	CL6	CL5	4	1.85	1.56	0.870167	
3	OB2	OB6	2	2.17	.	0.879348	
2	CL3	OB3	3	2.80	1.25	0.957954	
1	CL4	CL2	7	.	2.80	1.087015	

```
                     The SAS System                        34
                           10:54 Thursday, December 17, 1992
                 Average Linkage Cluster Analysis
                   Name of Observation or Cluster
        OB    OB    OB    OB    OB    OB    OB
        1     7     4     5     2     6     3
   1.2 +
       |
       |
       |
       |
       |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
       |XXXXXXXXXXXXXXXXXXX     XXXXXXXXXXXXX
       |XXXXXXXXXXXXXXXXXXX     XXXXXXXXXXXXX
A    1 +XXXXXXXXXXXXXXXXXXX     XXXXXXXXXXXXX
v      |XXXXXXXXXXXXXXXXXXX     XXXXXXXXXXXXX
e      |XXXXXXXXXXXXXXXXXXX     XXXXXXXXXXXXX
r      |XXXXXXXXXXXXXXXXXXX     XXXXXXX     .
a      |XXXXXXXXXXXXXXXXXXX     XXXXXXX     .
g      |XXXXXXXXXXXXXXXXXXX     XXXXXXX     .
e      |XXXXXXX     XXXXXXX     .     .     .
       |XXXXXXX     XXXXXXX     .     .     .
D  0.8 +XXXXXXX     XXXXXXX     .     .     .
i      |XXXXXXX     .     .     .     .     .
s      |XXXXXXX     .     .     .     .     .
t      |.     .     .     .     .     .     .
a      |.     .     .     .     .     .     .
n      |.     .     .     .     .     .     .
c      |.     .     .     .     .     .     .
e      |.     .     .     .     .     .     .
   0.6 +.     .     .     .     .     .     .
B      |.     .     .     .     .     .     .
e      |.     .     .     .     .     .     .
t      |.     .     .     .     .     .     .
w      |.     .     .     .     .     .     .
e      |.     .     .     .     .     .     .
e      |.     .     .     .     .     .     .
n      |.     .     .     .     .     .     .
   0.4 +.     .     .     .     .     .     .
C      |.     .     .     .     .     .     .
l      |.     .     .     .     .     .     .
u      |.     .     .     .     .     .     .
s      |.     .     .     .     .     .     .
t      |.     .     .     .     .     .     .
e      |.     .     .     .     .     .     .
r      |.     .     .     .     .     .     .
s  0.2 +.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
       |.     .     .     .     .     .     .
     0 +.     .     .     .     .     .     .
```

Ward's Minimum Variance Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	4024.21	2288.74	0.383053	0.38305
2	1735.47	47.18	0.165195	0.54825
3	1688.29	297.21	0.160704	0.70895
4	1391.08	360.47	0.132413	0.84137
5	1030.62	394.68	0.098101	0.93947
6	635.94	635.94	0.060533	1.00000
7	0.00	0.00	0.000000	1.00000
8	0.00	0.00	0.000000	1.00000
9	0.00	0.00	0.000000	1.00000
10	0.00	0.00	0.000000	1.00000
11	0.00	0.00	0.000000	1.00000
12	0.00	0.00	0.000000	1.00000
13	0.00	0.00	0.000000	1.00000
14	0.00	0.00	0.000000	1.00000
15	0.00	0.00	0.000000	1.00000
16	0.00	0.00	0.000000	1.00000
17	0.00	0.00	0.000000	1.00000
18	0.00	0.00	0.000000	1.00000
19	0.00	0.00	0.000000	1.00000
20	0.00	0.00	0.000000	1.00000
21	0.00	0.00	0.000000	1.00000
22	0.00	0.00	0.000000	1.00000
23	0.00	0.00	0.000000	1.00000
24	0.00	0.00	0.000000	1.00000
25	0.00	0.00	0.000000	1.00000
26	0.00	0.00	0.000000	1.00000
27	0.00	0.00	0.000000	1.00000
28	0.00	0.00	0.000000	1.00000
29	0.00	0.00	0.000000	1.00000
30	-0.00	0.00	-.000000	1.00000
31	-0.00	0.00	-.000000	1.00000
32	-0.00	0.00	-.000000	1.00000
33	-0.00	0.00	-.000000	1.00000
34	-0.00	0.00	-.000000	1.00000
35	-0.00	0.00	-.000000	1.00000
36	-0.00	0.00	-.000000	1.00000
37	-0.00	0.00	-.000000	1.00000
38	-0.00	0.00	-.000000	1.00000
39	-0.00	0.00	-.000000	1.00000
40	-0.00	0.00	-.000000	1.00000
41	-0.00	0.00	-.000000	1.00000
42	-0.00	0.00	-.000000	1.00000
43	-0.00	0.00	-.000000	1.00000
44	-0.00	0.00	-.000000	1.00000
45	-0.00	0.00	-.000000	1.00000
46	-0.00	0.00	-.000000	1.00000
47	-0.00	.	-.000000	1.00000

Root-Mean-Square Total-Sample Standard Deviation = 14.95071
Root-Mean-Square Distance Between Observations = 144.9525

Ward's Minimum Variance Cluster Analysis

NCL	Clusters Joined		FREQ	SPRSQ	RSQ	Pseudo F	Pseudo t**2	Tie
6	OB1	OB7	2	0.092038	0.907962	2.0	.	
5	OB4	OB5	2	0.105198	0.802764	2.0	.	
4	OB2	OB6	2	0.128875	0.673889	2.1	.	
3	CL6	CL5	4	0.153779	0.520110	2.2	1.6	
2	CL4	OB3	3	0.160969	0.359141	2.8	1.2	
1	CL3	CL2	7	0.359141	0.000000	.	2.8	

```
                                The SAS System                          37
                                        10:54 Thursday, December 17, 1992
          Ward's Minimum Variance Cluster Analysis
                Name of Observation or Cluster
          OB   OB   OB   OB   OB   OB   OB
          1    7    4    5    2    6    3
     0.4 +
         |
         |
         |
         |
         |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    0.35 +XXXXXXXXXXXXXXXX    XXXXXXXXXXX
         |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
         |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
         |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
         |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
         |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
     0.3 +XXXXXXXXXXXXXXXX    XXXXXXXXXXX
S        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
e        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
m        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
i        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
-        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
P   0.25 +XXXXXXXXXXXXXXXX    XXXXXXXXXXX
a        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
r        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
t        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
i        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
a        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
l    0.2 +XXXXXXXXXXXXXXXX    XXXXXXXXXXX
         |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
R        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
-        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
S        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
q        |XXXXXXXXXXXXXXXX    XXXXXXXXXXX
u   0.15 +XXXXXXXXXXXXXXXX    XXXXXX    .
a        |XXXXXX    XXXXXX    XXXXXX    .
r        |XXXXXX    XXXXXX    XXXXXX    .
e        |XXXXXX    XXXXXX    XXXXXX    .
d        |XXXXXX    XXXXXX    .    .    .
         |XXXXXX    XXXXXX    .    .    .
     0.1 +XXXXXX    .    .    .    .    .
         |XXXXXX    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
    0.05 +.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
         |.    .    .    .    .    .    .
       0 +.    .    .    .    .    .    .
```

OBS	LDERP	LDERT	SLTKP	SLTKT	SLLDP	SLLDT	FTP	FTT	FLP	FLT	SLP	SLT	STP	STT
1	1	31	0	57	1	33	0	60	1	59	1	42	1	37
2	1	40	0	53	1	47	1	35	0	60	1	38	1	60
3	0	60	1	57	1	27	1	40	0	60	1	18	1	60
4	1	60	1	60	0	60	1	60	1	30	1	60	1	60
5	1	60	0	60	1	60	1	34	0	60	1	37	1	30
6	1	60	1	32	0	60	1	60	0	60	1	40	1	16
7	0	22	0	52	1	26	0	60	0	60	1	40	1	14
8	0	60	1	58	1	30	1	20	1	19	1	60	0	60
9	0	60	0	60	0	60	0	60	0	60	0	60	0	60
10	1	60	1	50	0	60	1	39	0	60	0	60	1	43
11	1	35	1	33	0	31	1	34	1	57	1	27	1	17
12	1	28	1	30	1	43	1	33	1	33	1	31	1	27
13	1	60	1	27	1	28	0	60	0	60	0	60	0	60
14	0	18	0	60	1	22	0	60	0	60	1	57	0	60
15	1	60	1	31	1	34	1	23	0	60	1	35	1	38
16	0	26	0	60	1	33	0	60	0	39	0	60	1	18
17	1	30	0	30	1	28	1	30	0	60	1	17	0	60
18	1	38	1	56	1	26	1	42	1	36	1	33	0	60
19	1	60	1	30	1	28	1	20	1	33	1	24	1	18
20	1	60	1	40	1	28	1	26	0	60	1	56	1	14
21	1	60	1	30	1	38	1	31	0	60	1	30	1	16
22	1	29	1	46	1	47	1	42	1	50	1	30	1	18
23	0	14	1	20	1	26	1	28	1	49	0	19	1	15
24	0	19	1	30	1	26	1	24	1	31	1	39	1	18
25	0	60	1	57	1	27	1	31	0	60	0	60	1	39
26	1	60	1	60	1	39	1	60	1	52	1	47	0	60
27	1	25	1	30	1	24	0	60	1	60	0	60	1	17
28	1	30	1	24	1	29	1	12	0	39	1	54	0	60
29	0	19	0	60	1	60	1	37	1	23	1	11	1	20
30	1	30	1	35	1	34	1	37	0	60	1	39	0	60
31	0	60	1	39	1	47	0	46	0	60	1	21	1	17
32	1	23	1	38	1	24	1	19	1	40	1	26	1	17
33	0	58	1	27	1	36	1	25	0	60	1	32	1	16
34	1	60	1	56	1	29	1	22	0	38	1	20	1	22
35	1	60	0	60	1	17	1	28	0	60	0	60	1	18
36	1	25	1	25	1	30	1	20	1	25	1	17	1	60
37	1	16	1	26	1	26	1	8	0	60	1	40	0	60
38	1	60	1	19	1	27	1	28	1	27	1	10	1	12
39	0	60	1	50	1	28	1	47	1	41	1	17	1	19
40	1	19	1	26	1	37	1	20	0	9	1	42	0	20
41	1	29	1	60	1	42	1	35	0	60	1	20	1	19
42	0	60	1	29	1	34	1	51	0	60	1	21	0	60
43	0	37	1	28	1	34	0	9	1	41	0	9	1	14
44	1	28	1	44	0	60	1	23	1	30	0	54	0	60
45	1	60	1	26	1	38	1	33	1	25	1	44	0	60
46	1	27	0	31	1	38	1	53	1	12	1	37	1	19
47	1	28	1	27	1	26	1	20	1	21	1	9	1	15

Variable	N	Mean	Std Dev	Minimum	Maximum
LDERP	47	0.6808511	0.4711864	0	1.0000000
LDERT	47	42.2127660	17.4915336	14.0000000	60.0000000
SLTKP	47	0.7659574	0.4279763	0	1.0000000
SLTKT	47	41.2553191	14.2805250	19.0000000	60.0000000
SLLDP	47	0.8723404	0.3373181	0	1.0000000
SLLDT	47	35.8936170	12.1032407	17.0000000	60.0000000
FTP	47	0.8085106	0.3977271	0	1.0000000
FTT	47	36.2765957	15.7793235	8.0000000	60.0000000
FLP	47	0.4680851	0.5043749	0	1.0000000
FLT	47	46.3617021	15.9692987	9.0000000	60.0000000
SLP	47	0.7872340	0.4136881	0	1.0000000
SLT	47	36.6595745	16.7114600	9.0000000	60.0000000
STP	47	0.7021277	0.4622673	0	1.0000000
STT	47	34.9574468	20.1461588	12.0000000	60.0000000

Appendix C. *SLAM and Fortran Code*

SLAM Code

This section contains the SLAM network code used. The first model shown is the original model (Pritsker, 1986:240). Figure 33 shows the SLAM network diagram. The next seven models are the network portion of the modified model for each scenario. The associated Fortran code is shown in the next section. Figureon page 113 shows the network diagram for the LDER scenario model. The scenarios were:

- LDER - Load interarrival time, loading time, hauling time, dumping time, and return time unchanged. (Original Model with Modifications)
- SLTK - Hauling time, dumping time, and return time doubled for the third truck. (Slow Truck)
- SLLD - Loading time doubled for the first loader. (Slow Loader)
- FT - Hauling time, dumping time, and return time cut in half for all trucks. (Fast Trucks)
- FL - Loading times halved for both loaders. (Fast Loaders)
- SL - Load interarrival time multiplied by 2. (Slow Loads)
- ST - Hauling time, dumping time, and return time doubled for all trucks. (Slow Trucks)

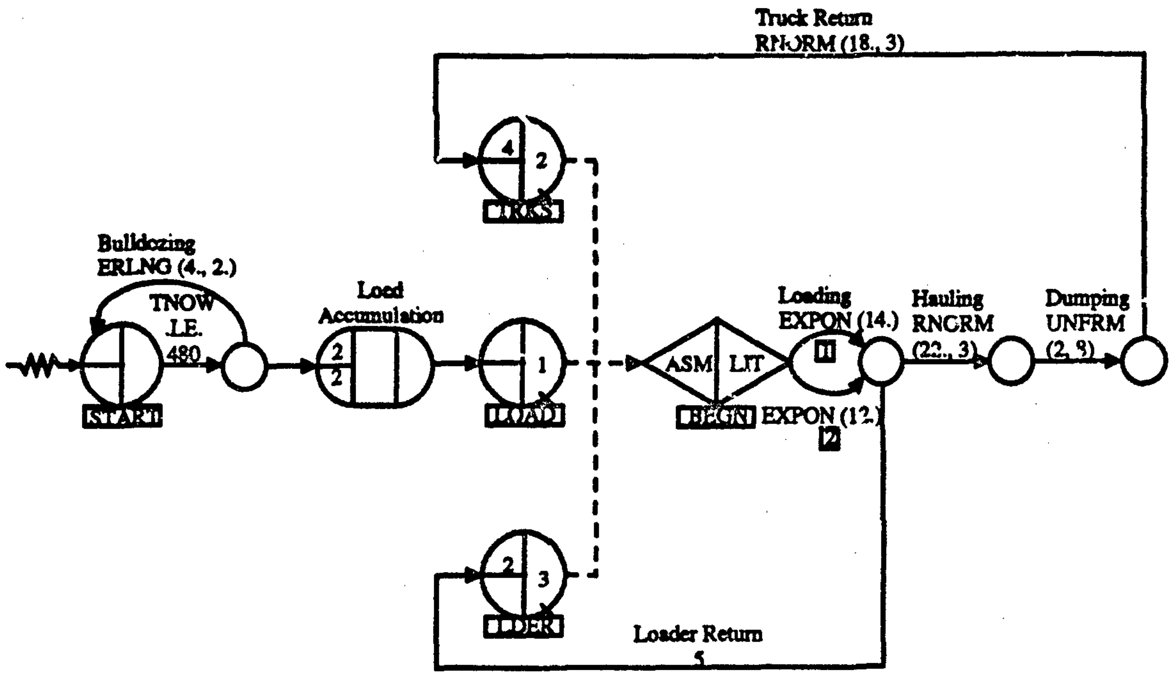


Figure 33. SLAM Network Diagram for Original Model.

Original model.

```
GEN,PRITSKER,TRUCK HAULING,7/12/83,1,,,Y/N;
LIMITS,3,1,50;
NETWORK;
START CREATE;                               CREATE LOAD TRANSACTIONS
      ACT,,TNOW.LE.480;                     STOP IF AFTER 8 HOURS
      GOON;                                 ELSE
      ACT,ERLNG(4.,2.),,START;              BRANCH BACK TO START
      ACT;                                  AND CONTINUE
      ACCUM,2,2;                            ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                     QUEUE OF LOADS
TRKS  QUEUE(2),4,,,BEGN;                    QUEUE OF TRUCKS
LDER  QUEUE(3),2,,,BEGN;                    QUEUE OF LOADERS
BEGN  SELECT,ASM,LIT,,LOAD,TRKS,LDER;       ASM OF LOAD,TRKS, AND LDER
      ACT/1,EXPON(14.);                     LOADER1 TIME
      ACT/2,EXPON(12.);                     LOADER2 TIME
      GOON;
      ACT,5,,LDER;                          LOADER RESTING TIME
      ACT,RNORM(22.,3.);                    TRUCK HAULING TIME
      GOON;
```

```
     ACT,UNFRM(2.,8.);                          TRUCK DUMPING TIME
     GOON;
     ACT,RNORM(18.,3.),,TRKS;                   TRUCK RETURN TIME
     END
FIN;
```

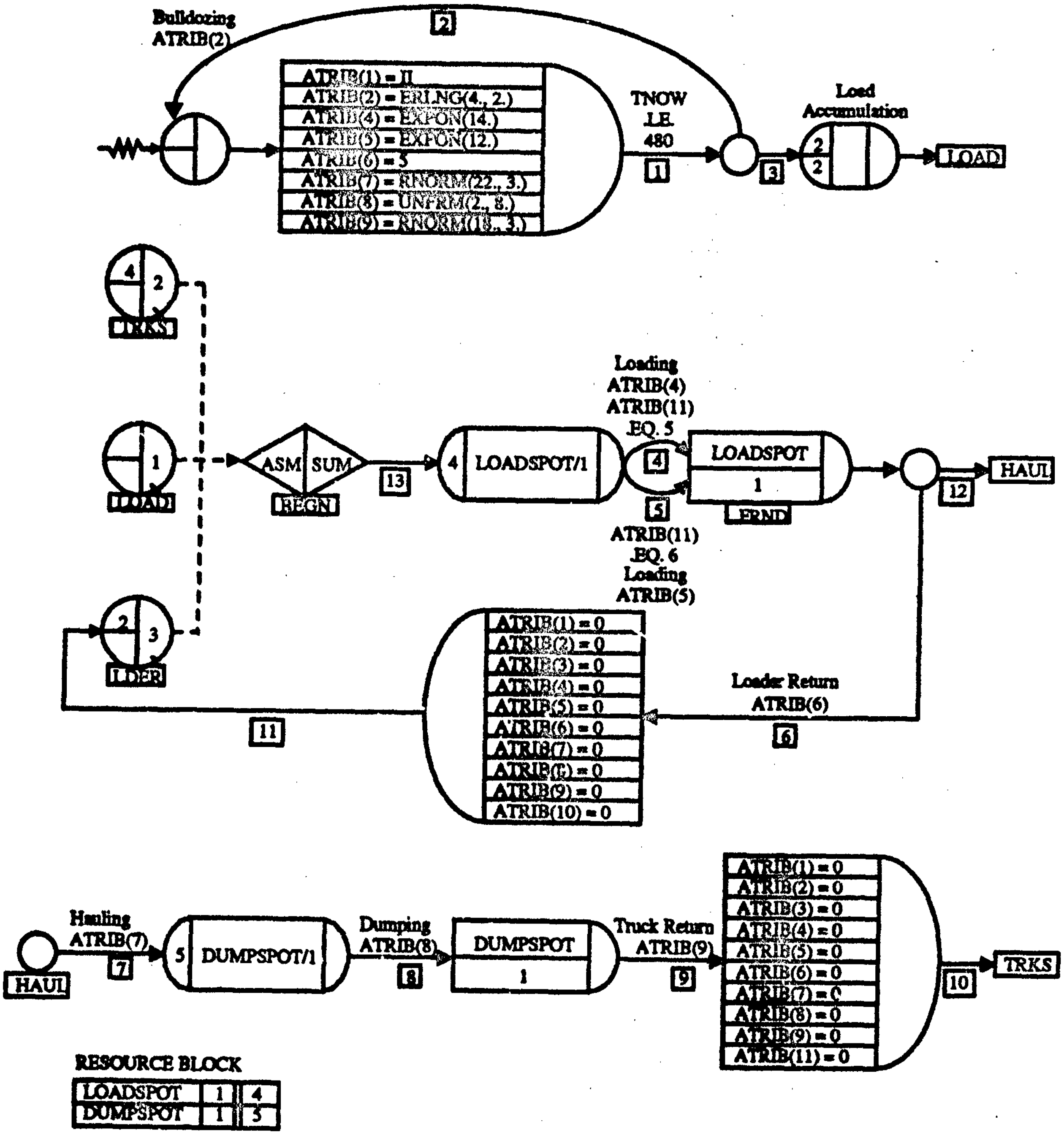


Figure 34. SLAM Network Diagram for LDER Model.

Modified model used for LDER scenario. All times were unchanged from the original model.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                     CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.),
             ATRIB(4) = EXPON(14.),
             ATRIB(5) = EXPON(12.),
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.),
             ATRIB(8) = UNFRM(2.,8.),
             ATRIB(9) = RNORM(18.,3.);
      ACT/1,,TNOW.LE.480;                         STOP IF AFTER 8 HOURS
      GOON;                                       ELSE
      ACT/2,ATRIB(2),,START;                      BRANCH BACK TO START
      ACT/3;                                       AND CONTINUE
      ACCUM,2,2;                                  ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                           QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                           QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                           QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;            ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;         LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;              LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,,HAUL;
      ACT/6,ATRIB(6);                             LOADER RESTING TIME
      ASSIGN,ATRIB(1)=0,
             ATRIB(2)=0,
             ATRIB(3)=0,
             ATRIB(4)=0,
             ATRIB(5)=0,
             ATRIB(6)=0,
```

```
            ATRIB(7)=0,
            ATRIB(8)=0,
            ATRIB(9)=0,
            ATRIB(10)=0;
      ACT/11,,,LDER;
HAUL  GOON;
      ACT/7,ATRIB(7);                          TRUCK HAULING TIME
      AWAIT(5),DUMPSPOT/1;
      ACT/8,ATRIB(8);                          TRUCK DUMPING TIME
      FREE,DUMPSPOT(1);
      ACT/9,ATRIB(9);                          TRUCK RETURN TIME
      ASSIGN,ATRIB(1)=0,
            ATRIB(2)=0,
            ATRIB(3)=0,
            ATRIB(4)=0,
            ATRIB(5)=0,
            ATRIB(6)=0,
            ATRIB(7)=0,
            ATRIB(8)=0,
            ATRIB(9)=0,
            ATRIB(11)=0;
      ACT/10,,,TRKS;
      END;
MONTR,TRACE,0,480;
FIN;
```

Modified model used for SLTK scenario. Attributes 7, 8, and 9 (hauling, dumping, and return time) were multiplied by 2 for the third truck. This was accomplished by branching on Attribute 10 (truck number) after the select node.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                  CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.),
             ATRIB(4) = EXPON(14.),
             ATRIB(5) = EXPON(12.),
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.),
             ATRIB(8) = UNFRM(2.,8.),
             ATRIB(9) = RNORM(18.,3.);
      ACT/1,,TNOW.LE.480;                      STOP IF AFTER 8 HOURS
      GOON;                                    ELSE
      ACT/2,ATRIB(2),,START;                   BRANCH BACK TO START
      ACT/3;                                     AND CONTINUE
      ACCUM,2,2;                               ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                        QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                        QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                        QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;         ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      GOON;
      ACT,,ATRIB(10).NE.3,LSND;
      ACT,,ATRIB(10).EQ.3;
      ASSIGN,ATRIB(7) = ATRIB(7) * 2.,
             ATRIB(8) = ATRIB(8) * 2.,
             ATRIB(9) = ATRIB(9) * 2.;
LSND  AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;      LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;           LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,HAUL;
```

```
         ACT/6,ATRIB(6);                               LOADER RESTING TIME
         ASSIGN,ATRIB(1)=0,
                ATRIB(2)=0,
                ATRIB(3)=0,
                ATRIB(4)=0,
                ATRIB(5)=0,
                ATRIB(6)=0,
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(10)=0;
         ACT/11,,,LDER;
    HAUL GOON;
         ACT/7,ATRIB(7);                               TRUCK HAULING TIME
         AWAIT(5),DUMPSPOT/1;
         ACT/8,ATRIB(8);                               TRUCK DUMPING TIME
         FREE,DUMPSPOT(1);
         ACT/9,ATRIB(9);                               TRUCK RETURN TIME
         ASSIGN,ATRIB(1)=0,
                ATRIB(2)=0,
                ATRIB(3)=0,
                ATRIB(4)=0,
                ATRIB(5)=0,
                ATRIB(6)=0,
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(11)=0;
         ACT/10,,,TRKS;
         END;
    MONTR,TRACE,0,480;
    FIN;
```

Modified model used for SLLD scenario. Attribute 4 (loading time for first loader) was multiplied by 2.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                    CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.),
             ATRIB(4) = EXPON(14.) * 2.,
             ATRIB(5) = EXPON(12.),
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.),
             ATRIB(8) = UNFRM(2.,8.),
             ATRIB(9) = RNORM(18.,3.);
      ACT/1,,TNOW.LE.480;                        STOP IF AFTER 8 HOURS
      GOON;                                      ELSE
      ACT/2,ATRIB(2),,START;                     BRANCH BACK TO START
      ACT/3;                                      AND CONTINUE
      ACCUM,2,2;                                 ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                          QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                          QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                          QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;           ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;        LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;             LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,,HAUL;
      ACT/6,ATRIB(6);                            LOADER RESTING TIME
      ASSIGN,ATRIB(1)=0,
             ATRIB(2)=0,
             ATRIB(3)=0,
             ATRIB(4)=0,
             ATRIB(5)=0,
             ATRIB(6)=0,
```

```
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(10)=0;
        ACT/11,,,LDER;
HAUL    GOON;
        ACT/7,ATRIB(7);                              TRUCK HAULING TIME
        AWAIT(5),DUMPSPOT/1;
        ACT/8,ATRIB(8);                              TRUCK DUMPING TIME
        FREE,DUMPSPOT(1);
        ACT/9,ATRIB(9);                              TRUCK RETURN TIME
        ASSIGN,ATRIB(1)=0,
                ATRIB(2)=0,
                ATRIB(3)=0,
                ATRIB(4)=0,
                ATRIB(5)=0,
                ATRIB(6)=0,
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(11)=0;
        ACT/10,,,TRKS;
        END;
MONTR,TRACE,0,480;
FIN;
```

Modified model used for FT scenario. Attributes 7, 8, and 9 (truck hauling, dumping, and return times) were cut in half for all the trucks.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                  CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.),
             ATRIB(4) = EXPON(14.),
             ATRIB(5) = EXPON(12.),
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.) * .5,
             ATRIB(8) = UNFRM(2.,8.) * .5,
             ATRIB(9) = RNORM(18.,3.) * .5;
      ACT/1,,TNOW.LE.480;                      STOP IF AFTER 8 HOURS
      GOON;                                    ELSE
      ACT/2,ATRIB(2),,START;                   BRANCH BACK TO START
      ACT/3;                                     AND CONTINUE
      ACCUM,2,2;                               ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                        QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                        QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                        QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;         ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;      LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;           LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,,HAUL;
      ACT/6,ATRIB(6);                          LOADER RESTING TIME
      ASSIGN,ATRIB(1)=0,
             ATRIB(2)=0,
             ATRIB(3)=0,
             ATRIB(4)=0,
             ATRIB(5)=0,
             ATRIB(6)=0,
```

```
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(10)=0;
         ACT/11,,,LDER;
HAUL     GOON;
         ACT/7,ATRIB(7);                        TRUCK HAULING TIME
         AWAIT(5),DUMPSPOT/1;
         ACT/8,ATRIB(8);                        TRUCK DUMPING TIME
         FREE,DUMPSPOT(1);
         ACT/9,ATRIB(9);                        TRUCK RETURN TIME
         ASSIGN,ATRIB(1)=0,
                ATRIB(2)=0,
                ATRIB(3)=0,
                ATRIB(4)=0,
                ATRIB(5)=0,
                ATRIB(6)=0,
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(11)=0;
         ACT/10,,,TRKS;
         END;
MONTR,TRACE,0,480;
FIN;
```

Modified model used for FL scenario. Attributes 4 and 5 (loader loading times) were multiplied by 0. 5.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                    CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.),
             ATRIB(4) = EXPON(14.) * .5,
             ATRIB(5) = EXPON(12.) * .5,
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.),
             ATRIB(8) = UNFRM(2.,8.),
             ATRIB(9) = RNORM(18.,3.);
      ACT/1,,TNOW.LE.480;                        STOP IF AFTER 8 HOURS
      GOON;                                      ELSE
      ACT/2,ATRIB(2),,START;                     BRANCH BACK TO START
      ACT/3;                                       AND CONTINUE
      ACCUM,2,2;                                 ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                          QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                          QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                          QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;           ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;        LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;             LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,,HAUL;
      ACT/6,ATRIB(6);                            LOADER RESTING TIME
      ASSIGN,ATRIB(1)=0,
             ATRIB(2)=0,
             ATRIB(3)=0,
             ATRIB(4)=0,
             ATRIB(5)=0,
             ATRIB(6)=0,
```

```
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(10)=0;
        ACT/11,,,LDER;
HAUL    GOON;
        ACT/7,ATRIB(7);                           TRUCK HAULING TIME
        AWAIT(5),DUMPSPOT/1;
        ACT/8,ATRIB(8);                           TRUCK DUMPING TIME
        FREE,DUMPSPOT(1);
        ACT/9,ATRIB(9);                           TRUCK RETURN TIME
        ASSIGN,ATRIB(1)=0,
                ATRIB(2)=0,
                ATRIB(3)=0,
                ATRIB(4)=0,
                ATRIB(5)=0,
                ATRIB(6)=0,
                ATRIB(7)=0,
                ATRIB(8)=0,
                ATRIB(9)=0,
                ATRIB(11)=0;
        ACT/10,,,TRKS;
        END;
MONTR,TRACE,0,480;
FIN;
```

Modified model used for SL scenario. Attribute 2 (load interarrival time) was doubled.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                    CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.) * 2.,
             ATRIB(4) = EXPON(14.),
             ATRIB(5) = EXPON(12.),
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.),
             ATRIB(8) = UNFRM(2.,8.),
             ATRIB(9) = RNORM(18.,3.);
      ACT/1,,TNOW.LE.480;                        STOP IF AFTER 8 HOURS
      GOON;                                      ELSE
      ACT/2,ATRIB(2),,START;                     BRANCH BACK TO START
      ACT/3;                                       AND CONTINUE
      ACCUM,2,2;                                 ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                          QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                          QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                          QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;           ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;        LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;             LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,,HAUL;
      ACT/6,ATRIB(6);                            LOADER RESTING TIME
      ASSIGN,ATRIB(1)=0,
             ATRIB(2)=0,
             ATRIB(3)=0,
             ATRIB(4)=0,
             ATRIB(5)=0,
             ATRIB(6)=0,
             ATRIB(7)=0,
```

```
                 ATRIB(8)=0,
                 ATRIB(9)=0,
                 ATRIB(10)=0;
        ACT/11,,,LDER;
HAUL    GOON;
        ACT/7,ATRIB(7);                           TRUCK HAULING TIME
        AWAIT(5),DUMPSPOT/1;
        ACT/8,ATRIB(8);                           TRUCK DUMPING TIME
        FREE,DUMPSPOT(1);
        ACT/9,ATRIB(9);                           TRUCK RETURN TIME
        ASSIGN,ATRIB(1)=0,
                 ATRIB(2)=0,
                 ATRIB(3)=0,
                 ATRIB(4)=0,
                 ATRIB(5)=0,
                 ATRIB(6)=0,
                 ATRIB(7)=0,
                 ATRIB(8)=0,
                 ATRIB(9)=0,
                 ATRIB(11)=0;
        ACT/10,,,TRKS;
        END;
MONTR,TRACE,0,480;
FIN;
```

Modified model used for ST scenario. Attributes 7, 8, and 9 (truck hauling, dumping, and return times) were multiplied by 2 for all trucks.

```
GEN,CARPENTER,TRUCK HAULING,9/30/92,1,,,Y/N,,,72;
LIMITS,5,14,50;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,1,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,2,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,3,0;
ENTRY/2,0,0,0,0,0,0,0,0,0,0,4,0;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,0,5;
ENTRY/3,0,0,0,0,0,0,0,0,0,0,0,6;
NETWORK;
      RESOURCE/LOADSPOT(1),4;
      RESOURCE/DUMPSPOT(1),5;
START CREATE;                                       CREATE LOAD TRANSACTIONS
      ASSIGN,ATRIB(1) = II,
             ATRIB(2) = ERLNG(4.,2.),
             ATRIB(4) = EXPON(14.),
             ATRIB(5) = EXPON(12.),
             ATRIB(6) = 5,
             ATRIB(7) = RNORM(22.,3.) * 2.,
             ATRIB(8) = UNFRM(2.,8.) * 2.,
             ATRIB(9) = RNORM(18.,3.) * 2.;
      ACT/1,,TNOW.LE.480;                           STOP IF AFTER 8 HOURS
      GOON;                                         ELSE
      ACT/2,ATRIB(2),,START;                        BRANCH BACK TO START
      ACT/3;                                          AND CONTINUE
      ACCUM,2,2;                                    ACCUMULATE TWO PILES
LOAD  QUEUE(1),,,,BEGN;                             QUEUE OF LOADS
TRKS  QUEUE(2),,,,BEGN;                             QUEUE OF TRUCKS
LDER  QUEUE(3),,,,BEGN;                             QUEUE OF LOADERS
BEGN  SELECT,ASM/SUM,,,LOAD,TRKS,LDER;              ASM OF LOAD,TRKS, AND LDER
      ACT/13;
      AWAIT(4),LOADSPOT/1,,1;
      ACT/4,ATRIB(4),ATRIB(11).EQ.5,FRND;           LOADER1 TIME
      ACT/5,ATRIB(5),ATRIB(11).EQ.6;                LOADER2 TIME
FRND  FREE,LOADSPOT(1);
      GOON;
      ACT/12,,,HAUL;
      ACT/6,ATRIB(6);                               LOADER RESTING TIME
      ASSIGN,ATRIB(1)=0,
             ATRIB(2)=0,
             ATRIB(3)=0,
             ATRIB(4)=0,
             ATRIB(5)=0,
             ATRIB(6)=0,
```

```
                    ATRIB(7)=0,
                    ATRIB(8)=0,
                    ATRIB(9)=0,
                    ATRIB(10)=0;
        ACT/11,,,LDER;
HAUL    GOON;
        ACT/7,ATRIB(7);                          TRUCK HAULING TIME
        AWAIT(5),DUMPSPOT/1;
        ACT/8,ATRIB(8);                          TRUCK DUMPING TIME
        FREE,DUMPSPOT(1);
        ACT/9,ATRIB(9);                          TRUCK RETURN TIME
        ASSIGN,ATRIB(1)=0,
                    ATRIB(2)=0,
                    ATRIB(3)=0,
                    ATRIB(4)=0,
                    ATRIB(5)=0,
                    ATRIB(6)=0,
                    ATRIB(7)=0,
                    ATRIB(8)=0,
                    ATRIB(9)=0,
                    ATRIB(11)=0;
        ACT/10,,,TRKS;
        END;
MONTR,TRACE,0,480;
FIN;
```

Fortran Code

This section contains the Fortran Code used in this experiment. These five sets of code were used with the SLAM network code to create the unsorted Proof trace files for the LDER scenario. The Fortran code for the other scenarios is the same except for different trace file names. The final set of code is the program used to sort the trace files.

```
C  Capt Mike Carpenter, GOR-93M, 6 Oct 92
C  This program generates the unsorted trace file for the color only
C  animation of the Truck Hauling simulation.  The program PRFSRTC must
C  be run to generate the sorted trace file (LDERC.ATF)
C
      PROGRAM MAIN
      DIMENSION NSET(10000)
      INCLUDE 'SLAM$DIR:PARAM.INC'
      COMMON/SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     +MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     +SSL(100),TNEXT, TNOW, XX(100)
      COMMON QSET(10000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=10000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      NPLOT=2
      OPEN(10,FILE='LDERC.TR',STATUS='UNKNOWN')
      CALL SLAM
      STOP
      END

C  Subroutine INTLC generates the loaders and the trucks and places
C  them on the screen.

      SUBROUTINE INTLC
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCOMM1/MYARRAY(5)
      XX(1) = 1000.
      WRITE(10,909)XX(1) * 0.0
      DO 10 I = 1,4
        IF (I .EQ. 1) THEN
          WRITE(10,901)I+4
901       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,902)I+4
902       FORMAT(1X,'PLACE ',I2,' ON Loader1')
        ENDIF
        IF (I .EQ. 2) THEN
          WRITE(10,905)I+4
905       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,906)I+4
906       FORMAT(1X,'PLACE ',I2,' ON Loader2')
        ENDIF
```

```
      WRITE(10,907)I
907   FORMAT(1X,'CREATE Truck ',I2)
      WRITE(10,908)I,I
908   FORMAT(1X,'PLACE ',I2,' ON Truck',I1)
      IF(I .NE. 4) THEN
        WRITE(10,909)I * 1.5
909     FORMAT(1X,'TIME ',F9.4)
      ENDIF
10    CONTINUE
      RETURN
      END

      SUBROUTINE OTPUT
      WRITE(10,10)
10    FORMAT(1X,'END')
      CLOSE(10)
      RETURN
      END

C  Subroutine UMONT is the user written SLAM trace that writes the
C  unsorted trace file after INTLC writes the statements to initialize
C  the animation.

      SUBROUTINE UMONT(IT)
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCONM1/MYARRAY(5)
      CHARACTER*4 LABL,NNLBL
      INTEGER NNUMB,ACTNUM,LOADNUM
      REAL LOADTIME
      IF(TNOW .EQ. 0.0) GO TO 999
      LABL=NNLBL(IDUM)
C  Returns current node label
      ACTNUM=NNUMB(IDUM)
C  Returns current activity number
      IF(IT .EQ. 0) GO TO 100
C  Go to 100 if a node label is hit
      IF(IT .EQ. -5) GO TO 999
C  UMONT code that is not needed

C  The following statements skip over activities that are not
C  important for the animation.

      IF(ACTNUM .EQ. -1) GO TO 999
      IF(ACTNUM .EQ. 1) GO TO 999
```

```
      IF(ACTNUM .EQ. 2) GO TO 999
      IF(ACTNUM .EQ. 3) GO TO 999
      IF(ACTNUM .EQ. 7) GO TO 999
      IF(ACTNUM .EQ. 10) GO TO 999
      IF(ACTNUM .EQ. 11) GO TO 999
      IF(ACTNUM .EQ. 12) GO TO 999
      IF(ACTNUM .EQ. 13) GO TO 999
      WRITE(10,900)TNOW
C  Write out current time
900   FORMAT(1X,'TIME ',F9.4)
      IF((ACTNUM .EQ. 4) .OR. (ACTNUM .EQ. 5)) THEN
C  Loading activites
        IF(ACTNUM .EQ. 4) THEN
C  Loader 1
          LOADTIME = ATRIB(4)/6.
C  Divide loading time by 6 to allow for state changes within
C  the animation
          LOADNUM = 5
        ENDIF
        IF(ACTNUM .EQ. 5) THEN
C  Loader 2
          LOADTIME = ATRIB(5)/6.
          LOADNUM = 6
        ENDIF
        ATRIB(11) = LOADNUM
        WRITE(10,800)LOADNUM
C  Traveling empty
800     FORMAT(1X,'SET ',I2,' COLOR GREEN')
        WRITE(10,805)TNOW + LOADTIME
805     FORMAT(1X,'TIME ',F9.4)
        WRITE(10,815)LOADNUM
C  Traveling loaded
815     FORMAT(1X,'SET ',I2,' COLOR RED')
        WRITE(10,805)TNOW + (3. * LOADTIME)
C  Increment time
        WRITE(10,800)LOADNUM
C  Traveling empty
        WRITE(10,830)ATRIB(10)
C  Truck partially loaded
830     FORMAT(1X,'SET ',F3.0,' COLOR PINK')
        WRITE(10,805)TNOW + (4. * LOADTIME)
C  Increment time
        WRITE(10,815)LOADNUM
C  Traveling loaded
      ENDIF
```

```
      IF(ACTNUM .EQ. 6) THEN
 C  Hauling activity
          LOADNUM = ATRIB(11)
          WRITE(10,800)LOADNUM
 C  Traveling empty
          WRITE(10,855)ATRIB(10)
 C  Truck traveling loaded
 855      FORMAT(1X,'SET ',F3.0,' COLOR RED')
          WRITE(10,805)TNOW + ATRIB(6)
 C  Increment time
          WRITE(10,880)LOADNUM
 C  Loader idle
 880      FORMAT(1X,'SET ',I2,' COLOR WHITE')
      ENDIF
      IF(ACTNUM .EQ. 8) THEN
 C  Dumping Activity
          WRITE(10,865)ATRIB(10)
 C  Truck dumping
 865      FORMAT(1X,'SET ',F3.0,' COLOR YELLOW')
      ENDIF
      IF(ACTNUM .EQ. 9) THEN
 C  Truck return activity
          WRITE(10,875)ATRIB(10)
 C  Truck traveling empty
 875      FORMAT(1X,'SET ',F3.0,' COLOR GREEN')
      ENDIF
      IT = 1
      RETURN

 C  The following statements skip over nodes that are not
 C  important for the animation.

 100  IF(LABL .EQ. '    ') GO TO 999
      IF(LABL .EQ. 'LOAD') GO TO 999
      IF(LABL .EQ. 'LDER') GO TO 999
      IF(LABL .EQ. 'BEGN') GO TO 999
      IF(LABL .EQ. 'HAUL') GO TO 999
      IF(LABL .EQ. 'FRND') GO TO 999
      IF(LABL .EQ. 'STAR') GO TO 999
      WRITE(10,900)TNOW
 C  Write out current time
      IF(LABL .EQ. 'TRKS') THEN
        WRITE(10,915)ATRIB(10)
 C  Truck idle
 915    FORMAT(1X,'SET ',F3.0,' COLOR WHITE')
```

```
      ENDIF
999   IT = 1
      RETURN
      END
```

```
C  Capt Mike Carpenter, GOR-93M, 6 Oct 92
C  This program generates the unsorted trace file for the color and icon
C  animation of the Truck Hauling simulation.  The program PRFSRTCI must
C  be run to generate the sorted trace file (LDERCI.ATF)
C
      PROGRAM MAIN
      DIMENSION NSET(10000)
      INCLUDE 'SLAM$DIR:PARAM.INC'
      COMMON/SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     +MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     +SSL(100),TNEXT, TNOW, XX(100)
      COMMON QSET(10000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=10000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      NPLOT=2
      OPEN(10,FILE='LDERCI.TR',STATUS='UNKNOWN')
      CALL SLAM
      STOP
      END

C  Subroutine INTLC generates the loaders and the trucks and places
C  them on the screen.

      SUBROUTINE INTLC
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCOMM1/MYARRAY(5)
      XX(1) = 1000.
      WRITE(10,909)XX(1) * 0.0
      DO 10 I = 1,4
        IF (I .EQ. 1) THEN
          WRITE(10,901)I+4
901       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,902)I+4
902       FORMAT(1X,'PLACE ',I2,' ON Loader1')
        ENDIF
        IF (I .EQ. 2) THEN
          WRITE(10,905)I+4
905       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,906)I+4
906       FORMAT(1X,'PLACE ',I2,' ON Loader2')
        ENDIF
```

```
      WRITE(10,907)I
907   FORMAT(1X,'CREATE Truck ',I2)
      WRITE(10,908)I,I
908   FORMAT(1X,'PLACE ',I2,' ON Truck',I1)
      IF(I .NE. 4) THEN
        WRITE(10,909)I * 1.5
909     FORMAT(1X,'TIME ',F9.4)
      ENDIF
10    CONTINUE
      RETURN
      END

      SUBROUTINE OTPUT
      WRITE(10,10)
10    FORMAT(1X,'END')
      CLOSE(10)
      RETURN
      END

C Subroutine UMONT is the user written SLAM trace that writes the
C unsorted trace file after INTLC writes the statements to initialize
C the animation.

      SUBROUTINE UMONT(IT)
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLKR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCOMM1/MYARRAY(5)
      CHARACTER*4 LABL,NNLBL
      INTEGER NNUMB,ACTNUM,LOADNUM
      REAL LOADTIME
      IF(TNOW .EQ. 0.0) GO TO 999
      LABL=NNLBL(IDUM)
C Returns current node label
      ACTNUM=NNUMB(IDUM)
C Returns current activity number
      IF(IT .EQ. 0) GO TO 100
C Go to 100 if at a node
      IF(IT .EQ. -5) GO TO 999
C UMONT code not needed for animation

C The following statements skip over activities that are not
C important for the animation.

      IF(ACTNUM .EQ. -1) GO TO 999
      IF(ACTNUM .EQ. 1) GO TO 999
```

```
      IF(ACTNUM .EQ. 2) GO TO 999
      IF(ACTNUM .EQ. 3) GO TO 999
      IF(ACTNUM .EQ. 7) GO TO 999
      IF(ACTNUM .EQ. 10) GO TO 999
      IF(ACTNUM .EQ. 11) GO TO 999
      IF(ACTNUM .EQ. 12) GO TO 999
      IF(ACTNUM .EQ. 13) GO TO 999
      WRITE(10,900)TNOW
C  Write out current time
900   FORMAT(1X,'TIME ',F9.4)
      IF((ACTNUM .EQ. 4) .OR. (ACTNUM .EQ. 5)) THEN
        IF(ACTNUM .EQ. 4) THEN
C  Loader 1
          LOADTIME = ATRIB(4)/6.
          LOADNUM = 5
        ENDIF
        IF(ACTNUM .EQ. 5) THEN
C  Loader 2
          LOADTIME = ATRIB(5)/6.
          LOADNUM = 6
        ENDIF
        ATRIB(11) = LOADNUM
        WRITE(10,800)LOADNUM
C  Traveling empty
800     FORMAT(1X,'SET ',I2,' COLOR GREEN')
        WRITE(10,805)TNOW + LOADTIME
C  Increment time
805     FORMAT(1X,'TIME ',F9.4)
        WRITE(10,814)LOADNUM
C  Change icon to Loaded
814     FORMAT(1X,'SET ',I2,' CLASS Loaded')
        WRITE(10,815)LOADNUM
C  Traveling loaded
815     FORMAT(1X,'SET ',I2,' COLOR RED')
        WRITE(10,805)TNOW + (3. * LOADTIME)
C  Increment time
        WRITE(10,825)LOADNUM
C  Change icon to Unloaded
        WRITE(10,800)LOADNUM
C  Traveling empty
825     FORMAT(1X,'SET ',I2,' CLASS Loader')
        WRITE(10,829)ATRIB(10)
C  Change truck icon to 1 load
829     FORMAT(1X,'SET ',F3.0,' CLASS Tload1')
        WRITE(10,830)ATRIB(10)
```

```
C  Traveling partially loaded
830     FORMAT(1X,'SET ',F3.0,' COLOR PINK')
        WRITE(10,805)TNOW + (4. * LOADTIME)
C  Increment time
        WRITE(10,814)LOADNUM
C  Change icon to Loaded
        WRITE(10,815)LOADNUM
C  Traveling loaded
      ENDIF
      IF(ACTNUM .EQ. 6) THEN
        LOADNUM = ATRIB(11)
        WRITE(10,325)LOADNUM
C  Change icon to Unloaded
        WRITE(10,800)LOADNUM
C  Traveling empty
        WRITE(10,854)ATRIB(10)
C  Change truck icon loaded
854     FORMAT(1X,'SET ',F3.0,' CLASS Tload2')
        WRITE(10,855)ATRIB(10)
C  Truck traveling loaded
855     FORMAT(1X,'SET ',F3.0,' COLOR RED')
        WRITE(10,805)TNOW + ATRIB(6)
C  Increment time
        WRITE(10,880)LOADNUM
C  Loader idle
880     FORMAT(1X,'SET ',I2,' COLOR WHITE')
      ENDIF
      IF(ACTNUM .EQ. 8) THEN
        WRITE(10,829)ATRIB(10)
C  Truck partially loaded
        WRITE(10,865)ATRIB(10)
C  Truck dumping
865     FORMAT(1X,'SET ',F3.0,' COLOR YELLOW')
        WRITE(10,805)TNOW + (ATRIB(8)/2.)
C  Increment time
        WRITE(10,864)ATRIB(10)
C  Change truck icon unloaded
864     FORMAT(1X,'SET ',F3.0,' CLASS Truck')
        WRITE(10,865)ATRIB(10)
C  Keep truck yellow
      ENDIF
      IF(ACTNUM .EQ. 9) THEN
        WRITE(10,875)ATRIB(10)
C  Truck traveling empty
875     FORMAT(1X,'SET ',F3.0,' COLOR GREEN')
```

```
      ENDIF
      IT = 1
      RETURN

C  The following statements skip over nodes that are not
C  important for the animation.

100   IF(LABL .EQ. '    ') GO TO 999
      IF(LABL .EQ. 'LOAD') GO TO 999
      IF(LABL .EQ. 'LDER') GO TO 999
      IF(LABL .EQ. 'BEGN') GO TO 999
      IF(LABL .EQ. 'HAUL') GO TO 999
      IF(LABL .EQ. 'FRND') GO TO 999
      IF(LABL .EQ. 'STAR') GO TO 999
      WRITE(10,900)TNOW
C  Write out current time
      IF(LABL .EQ. 'TRKS') THEN
        WRITE(10,915)ATRIB(10)
C  Truck idle
915     FORMAT(1X,'SET ',F3.0,' COLOR WHITE')
      ENDIF
999   IT = 1
      RETURN
      END
```

```
C  Capt Mike Carpenter, GOR-93M, 6 Oct 92
C  This program generates the unsorted trace file for the icon only
C  animation of the Truck Hauling simulation.  The program PRFSRTI must
C  be run to generate the sorted trace file (LDERI.ATF)
C
      PROGRAM MAIN
      DIMENSION NSET(10000)
      INCLUDE 'SLAM$DIR:PARAM.INC'
      COMMON/SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     +MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     +SSL(100),TNEXT, TNOW, XX(100)
      COMMON QSET(10000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=10000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      NPLOT=2
      OPEN(10,FILE='LDERI.TR',STATUS='UNKNOWN')
      CALL SLAM
      STOP
      END

C  Subroutine INTLC generates the loaders and the trucks and places
C  them on the screen.

      SUBROUTINE INTLC
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCOMM1/MYARRAY(5)
      XX(1) = 1000.
      WRITE(10,909)XX(1) * 0.0
      DO 10 I = 1,4
        IF (I .EQ. 1) THEN
          WRITE(10,901)I+4
901       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,902)I+4
902       FORMAT(1X,'PLACE ',I2,' ON Loader1')
        ENDIF
        IF (I .EQ. 2) THEN
          WRITE(10,905)I+4
905       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,906)I+4
906       FORMAT(1X,'PLACE ',I2,' ON Loader2')
        ENDIF
```

```
         WRITE(10,907)I
907      FORMAT(1X,'CREATE Truck ',I2)
         WRITE(10,908)I,I
908      FORMAT(1X,'PLACE ',I2,' ON Truck',I1)
         IF(I .NE. 4) THEN
           WRITE(10,909)I * 1.5
909        FORMAT(1X,'TIME ',F9.4)
         ENDIF
10     CONTINUE
       RETURN
       END

       SUBROUTINE OTPUT
       WRITE(10,10)
10     FORMAT(1X,'END')
       CLOSE(10)
       RETURN
       END

C  Subroutine UMONT is the user written SLAM trace that writes the
C  unsorted trace file after INTLC writes the statements to initialize
C  the animation.

       SUBROUTINE UMONT(IT)
       COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
      1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
       COMMON/UCOMM1/MYARRAY(5)
       CHARACTER*4 LABL,NNLBL
       INTEGER NNUMB,ACTNUM,LOADNUM
       REAL LOADTIME
       IF(TNOW .EQ. 0.0) GO TO 999
       LABL=NNLBL(IDUM)
C  Returns current node label
       ACTNUM=NNUMB(IDUM)
C  Returns current activity number
       IF(IT .EQ. 0) GO TO 999
C  If at node return
       IF(IT .EQ. -5) GO TO 999
C  UMONT code not needed

C  The following statements skip over activities that are not
C  important for the animation.

       IF(ACTNUM .EQ. -1) GO TO 999
       IF(ACTNUM .EQ. 1) GO TO 999
```

```
      IF(ACTNUM .EQ. 2) GO TO 999
      IF(ACTNUM .EQ. 3) GO TO 999
      IF(ACTNUM .EQ. 7) GO TO 999
      IF(ACTNUM .EQ. 9) GO TO 999
      IF(ACTNUM .EQ. 10) GO TO 999
      IF(ACTNUM .EQ. 11) GO TO 999
      IF(ACTNUM .EQ. 12) GO TO 999
      IF(ACTNUM .EQ. 13) GO TO 999
      WRITE(10,900)TNOW
C  Write out current time
900   FORMAT(1X,'TIME ',F9.4)
      IF((ACTNUM .EQ. 4) .OR. (ACTNUM .EQ. 5)) THEN
C  If loader activity
        IF(ACTNUM .EQ. 4) THEN
C  Loader 1
          LOADTIME = ATRIB(4)/6.
          LOADNUM = 5
        ENDIF
        IF(ACTNUM .EQ. 5) THEN
C  Loader 2
          LOADTIME = ATRIB(5)/6.
          LOADNUM = 6
        ENDIF
        ATRIB(11) = LOADNUM
        WRITE(10,805)TNOW + LOADTIME
C  Increment time
805     FORMAT(1X,'TIME ',F9.4)
        WRITE(10,815)LOADNUM
C  Change loader icon to Loaded
815     FORMAT(1X,'SET ',I2,' CLASS Loaded')
        WRITE(10,805)TNOW + (3. * LOADTIME)
C  Increment time
        WRITE(10,825)LOADNUM
C  Change loader icon to Unloaded
825     FORMAT(1X,'SET ',I2,' CLASS Loader')
        WRITE(10,830)ATRIB(10)
C  Change truck icon partially loaded
830     FORMAT(1X,'SET ',F3.0,' Class Tloadi')
        WRITE(10,805)TNOW + (4. * LOADTIME)
C  Increment time
        WRITE(10,815)LOADNUM
C  Change loader icon to Loaded
      ENDIF
      IF(ACTNUM .EQ. 6) THEN
C  Hauling activity
```

```
      LOADNUM = ATRIB(11)
      WRITE(10,855)ATRIB(10)
C  Change truck icon Loaded
855   FORMAT(1X,'SET ',F3.0,' CLASS Tload2')
      WRITE(10,825)LOADNUM
C  Change loader icon to Unloaded
     ENDIF
     IF(ACTNUM .EQ. 8) THEN
C  Dumping activity
      WRITE(10,830)ATRIB(10)
C  Change truck partially loaded
      WRITE(10,805)TNOW + (ATRIB(8)/2.)
C  Increment time
      WRITE(10,865)ATRIB(10)
C  Change truck to empty
865   FORMAT(1X,'SET ',F3.0,' CLASS Truck')
     ENDIF
999  IT = 1
     RETURN
     END
```

```
C  Capt Mike Carpenter, GOR-93M, 6 Oct 92
C  This program generates the unsorted trace file for the movement and
C  color and the movement, icon, and color
C  animations of the Truck Hauling simulation.  The program PRFSRTMC
C  must be run to generate the sorted trace files
C  (LDERMC.ATF and LDERMCI.ATF)
C
      PROGRAM MAIN
      DIMENSION NSET(10000)
      INCLUDE 'SLAM$DIR:PARAM.INC'
      COMMON/SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     +MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     +SSL(100),TNEXT, TNOW, XX(100)
      COMMON QSET(10000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=10000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      NPLOT=2
      OPEN(10,FILE='LDERMC.TR',STATUS='UNKNOWN')
      CALL SLAM
      STOP
      END

C  Subroutine INTLC generates the loaders and the trucks and places
C  them on the screen.

      SUBROUTINE INTLC
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCONM1/MYARRAY(5)
      II = 6
      XX(1) = 1000.
      WRITE(10,909)XX(1) * 0.0
      DO 10 I = 1,4
        IF (I .EQ. 1) THEN
          II = II + 1
          WRITE(10,901)I+4
901       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,902)I+4
902       FORMAT(1X,'PLACE ',I2,' ON Getload1')
          WRITE(10,903)II
903       FORMAT(1X,'CREATE Load ',I3)
          WRITE(10,904)II
```

```
904       FORMAT(1X,'PLACE ',I3,' ON Loadpath')
        ENDIF
        IF (I .EQ. 2) THEN
          WRITE(10,905)I+4
905       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,906)I+4
906       FORMAT(1X,'PLACE ',I2,' ON Getload2')
        ENDIF
        WRITE(10,907)I
907     FORMAT(1X,'CREATE Truck ',I2)
        WRITE(10,908)I
908     FORMAT(1X,'PLACE ',I2,' ON Truckload')
        IF(I .NE. 4) THEN
          WRITE(10,909)I * 1.5
909       FORMAT(1X,'TIME ',F9.4)
        ENDIF
10    CONTINUE
      RETURN
      END

      SUBROUTINE OTPUT
      WRITE(10,10)
10    FORMAT(1X,'END')
      CLOSE(10)
      RETURN
      END

C  Subroutine UMONT is the user written SLAM trace that writes the
C  unsorted trace file after INTLC writes the statements to initialize
C  the animation.

      SUBROUTINE UMONT(IT)
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCONM1/MYARRAY(5)
      CHARACTER*4 LABL,NNLBL
      INTEGER NNUMB,ACTNUM,LOADNUM
      REAL LOADTIME
      IF(TNOW .EQ. 0.0) GO TO 999
      LABL=NNLBL(IDUM)
      ACTNUM=NNUMB(IDUM)
      IF(IT .EQ. 0) GO TO 100
      IF(IT .EQ. -5) GO TO 999

C  The following statements skip over activities that are not
```

```
C  important for the animation.

      IF(ACTNUM .EQ. -1) GO TO 999
      IF(ACTNUM .EQ. 1) GO TO 999
      IF(ACTNUM .EQ. 2) GO TO 999
      IF(ACTNUM .EQ. 3) GO TO 999
      IF(ACTNUM .EQ. 7) GO TO 999
      IF(ACTNUM .EQ. 10) GO TO 999
      IF(ACTNUM .EQ. 11) GO TO 999
      IF(ACTNUM .EQ. 12) GO TO 999
      IF(ACTNUM .EQ. 13) GO TO 999
      WRITE(10,900)TNOW
C  Write out current time
900   FORMAT(1X,'TIME ',F9.4)
      IF((ACTNUM .EQ. 4) .OR. (ACTNUM .EQ. 5)) THEN
C Loading activity
        IF(ACTNUM .EQ. 4) THEN
C  Loader 1
          LOADTIME = ATRIB(4)/6.
          LOADNUM = 5
          WRITE(10,790)LOADNUM
C  Send Loader to get load
790       FORMAT(1X,'PLACE ',I2,' ON Getload1')
        ENDIF
        IF(ACTNUM .EQ. 5) THEN
C  Loader 2
          LOADTIME = ATRIB(5)/6.
          LOADNUM = 6
          WRITE(10,795)LOADNUM
C  Send Loader to get load
795       FORMAT(1X,'PLACE ',I2,' ON Getload2')
        ENDIF
        ATRIB(11) = LOADNUM
        WRITE(10,799)LOADNUM
C  Traveling empty
799     FORMAT(1X,'SET ',I2,' COLOR GREEN')
        WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
800     FORMAT(1X,'SET ',I2,' TRAVEL ',F9.4)
        WRITE(10,805)TNOW + LOADTIME
C  Increment time
805     FORMAT(1X,'TIME ',F9.4)
        WRITE(10,810)ATRIB(1) - 1
C  Loader picks up load
810     FORMAT(1X,'DESTROY ',F3.0)
```

```
      WRITE(10,815)LOADNUM
C  Change loader to Loaded
815   FORMAT(1X,'SET ',I2,' CLASS Loaded')
      WRITE(10,816)LOADNUM
C  Traveling loaded
816   FORMAT(1X,'SET ',I2,' COLOR RED')
      WRITE(10,820)LOADNUM
C  Place load in truck
820   FORMAT(1X,'PLACE ',I2,' ON Dumpload')
      WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
      WRITE(10,805)TNOW + (3. * LOADTIME)
C  Increment time
      WRITE(10,825)LOADNUM
C  Change loader icon to unloaded
825   FORMAT(1X,'SET ',I2,' CLASS Loader')
      WRITE(10,799)LOADNUM
C  Traveling empty
      WRITE(10,830)ATRIB(10)
C  Change truck icon to partially loaded
830   FORMAT(1X,'SET ',F3.0,' CLASS Tload1')
      WRITE(10,831)ATRIB(10)
C  Truck partially loaded
831   FORMAT(1X,'SET ',F3.0,' COLOR PINK')
      WRITE(10,835)LOADNUM
C  Send loader to get new load
835   FORMAT(1X,'PLACE ',I2,' ON Getnewload')
      WRITE(10,800)LOADNUM,LOADTIME
C  Set loader travel time
      WRITE(10,805)TNOW + (4. * LOADTIME)
C  Increment time
      WRITE(10,810)ATRIB(1)
C  Loader picks up second load
      WRITE(10,815)LOADNUM
C  Change loader to loaded
      WRITE(10,816)LOADNUM
C  Traveling loaded
      WRITE(10,820)LOADNUM
C  Place load in truck
      WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
    ENDIF
    IF(ACTNUM .EQ. 6) THEN
C  Hauling activity
      LOADNUM = ATRIB(11)
```

```
      WRITE(10,825)LOADNUM
C  Change loader icon to unloaded
      WRITE(10,799)LOADNUM
C  Traveling empty
      IF(LOADNUM .EQ. 5) THEN
        WRITE(10,840)LOADNUM
C  Return loader 1
840     FORMAT(1X,'PLACE ',I2,' ON Return1')
      ENDIF
      IF(LOADNUM .EQ. 6) THEN
        WRITE(10,845)LOADNUM
C  Return loader 2
845     FORMAT(1X,'PLACE ',I2,' ON Return2')
      ENDIF
      WRITE(10,850)LOADNUM,ATRIB(6)
C  Set travel time
850   FORMAT(1X,'SET ',I2,' TRAVEL ',F3.0)
      WRITE(10,855)ATRIB(10)
C  Change truck to loaded
855   FORMAT(1X,'SET ',F3.0,' CLASS Tload2')
      WRITE(10,856)ATRIB(10)
C  Truck traveling loaded
856   FORMAT(1X,'SET ',F3.0,' COLOR RED')
      WRITE(10,860)ATRIB(10)
C  Start truck hauling
860   FORMAT(1X,'PLACE ',F3.0,' ON Truckdump')
      WRITE(10,875)ATRIB(10),ATRIB(7)
C  Set truck travel time
      WRITE(10,805)TNOW + ATRIB(6)
C  Increment time
      WRITE(10,880)LOADNUM
C  Loader idle
880   FORMAT(1X,'SET ',I2,' COLOR WHITE')
     ENDIF
     IF(ACTNUM .EQ. 8) THEN
C  Dumping activity
      XX(1) = XX(1) + 1
      WRITE(10,861)ATRIB(10)
C  Start truck dumping
861   FORMAT(1X,'PLACE ',F3.0,' ON Dumping')
      WRITE(10,862)ATRIB(10),ATRIB(8)/2.
C  Set travel time
862   FORMAT(1X,'SET ',F3.0,' TRAVEL ',F9.4)
      WRITE(10,805)TNOW + (ATRIB(8)/2.)
C  Increment time
```

```
      WRITE(10,865)ATRIB(10)
C  Change truck to empty
865   FORMAT(1X,'SET ',F3.0,' CLASS Truck')
      WRITE(10,863)ATRIB(10)
C  Truck dumping
863   FORMAT(1X,'SET ',F3.0,' COLOR YELLOW')
      WRITE(10,866)XX(1)
C  Place two loads in truck
866   FORMAT(1X,'CREATE Twoloads ',F5.0)
      WRITE(10,867)XX(1)
C  Start two loads dumping
867   FORMAT(1X,'PLACE ',F5.0,' ON Loaddump')
     ENDIF
     IF(ACTNUM .EQ. 9) THEN
C  Truck return activity
      WRITE(10,869)ATRIB(10)
C  Truck traveling empty
869   FORMAT(1X,'SET ',F3.0,' COLOR GREEN')
      WRITE(10,870)ATRIB(10)
C  Start truck returning
870   FORMAT(1X,'PLACE ',F3.0,' ON Truckret')
      WRITE(10,875)ATRIB(10),ATRIB(9)
C  Set truck travel time
875   FORMAT(1X,'SET ',F3.0,' TRAVEL ',F9.4)
     ENDIF
     IT = 1
     RETURN

C  The following statements skip over nodes that are not
C  important for the animation.

100  IF(LABL .EQ. '    ') GO TO 999
     IF(LABL .EQ. 'LOAD') GO TO 999
     IF(LABL .EQ. 'LDER') GO TO 999
     IF(LABL .EQ. 'BEGN') GO TO 999
     IF(LABL .EQ. 'HAUL') GO TO 999
     IF(LABL .EQ. 'FRND') GO TO 999
     WRITE(10,900)TNOW
C  Write out current time
     IF(LABL .EQ. 'STAR') THEN
      II = II + 1
      WRITE(10,905)II
C  Create new load
905   FORMAT(1X,'CREATE Load ',I3)
      WRITE(10,910)II
```

```
C  Send load for pickup
910      FORMAT(1X,'PLACE ',I3,' ON Loadpath')
      ENDIF
      IF(LABL .EQ. 'TRKS') THEN
        WRITE(10,911)ATRIB(10)
C  Truck idle
911      FORMAT(1X,'SET ',F3.0,' COLOR WHITE')
        WRITE(10,915)ATRIB(10)
C  Place truck in loading queue
915      FORMAT(1X,'PLACE ',F3.0,' ON Truckload')
        WRITE(10,920)ATRIB(10)
C  Set truck travel time
920      FORMAT(1X,'SET ',F3.0,' TRAVEL 2')
      ENDIF
999   IT = 1
      RETURN
      END
```

```
C  Capt Mike Carpenter, GOR-93M, 6 Oct 92
C  This program generates the unsorted trace file for the movement only
C  and the movement and icon animations of the Truck Hauling simulation.
C  The program PRFSRTMI must be run to generate the sorted trace files
C  (LDERM.ATF and LDERMI.ATF)
C
      PROGRAM MAIN
      DIMENSION NSET(10000)
      INCLUDE 'SLAM$DIR:PARAM.INC'
      COMMON/SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     +MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     +SSL(100),TNEXT, TNOW, XX(100)
      COMMON QSET(10000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=10000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      NPLOT=2
      OPEN(10,FILE='LDERMI.TR',STATUS='UNKNOWN')
      CALL SLAM
      STOP
      END

C  Subroutine INTLC generates the loaders and the trucks and places
C  them on the screen.

      SUBROUTINE INTLC
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCONM1/MYARRAY(5)
      II = 6
      XX(1) = 1000.
      WRITE(10,909)XX(1) * 0.0
      DO 10 I = 1,4
        IF (I .EQ. 1) THEN
          II = II + 1
          WRITE(10,901)I+4
901       FORMAT(1X,'CREATE Loader ',I2)
          WRITE(10,902)I+4
902       FORMAT(1X,'PLACE ',I2,' ON Getload1')
          WRITE(10,903)II
903       FORMAT(1X,'CREATE Load ',I3)
          WRITE(10,904)II
904       FORMAT(1X,'PLACE ',I3,' ON Loadpath')
```

```
      ENDIF
      IF (I .EQ. 2) THEN
        WRITE(10,905)I+4
905     FORMAT(1X,'CREATE Loader ',I2)
        WRITE(10,906)I+4
906     FORMAT(1X,'PLACE ',I2,' ON Getload2')
      ENDIF
      WRITE(10,907)I
907   FORMAT(1X,'CREATE Truck ',I2)
      WRITE(10,908)I
908   FORMAT(1X,'PLACE ',I2,' ON Truckload')
      IF(I .NE. 4) THEN
        WRITE(10,909)I * 1.5
909     FORMAT(1X,'TIME ',F9.4)
      ENDIF
10    CONTINUE
      RETURN
      END

      SUBROUTINE OTPUT
      WRITE(10,10)
10    FORMAT(1X,'END')
      CLOSE(10)
      RETURN
      END

C  Subroutine UMONT is the user written SLAM trace that writes the
C  unsorted trace file after INTLC writes the statements to initialize
C  the animation.

      SUBROUTINE UMONT(IT)
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLMR
     1,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/UCONM1/MYARRAY(5)
      CHARACTER*4 LABL,NNLBL
      INTEGER NNUMB,ACTNUM,LOADNUM
      REAL LOADTIME
      IF(TNOW .EQ. 0.0) GO TO 999
      LABL=NNLBL(IDUM)
C  Returns current node label
      ACTNUM=NNUMB(IDUM)
C  Returns current activity number
      IF(IT .EQ. 0) GO TO 100
C  Go to 100 if at node
      IF(IT .EQ. -5) GO TO 999
```

```
C  UMONT code that is not needed

C  The following statements skip over activities that are not
C  important for the animation.

      IF(ACTNUM .EQ. -1) GO TO 999
      IF(ACTNUM .EQ. 1) GO TO 999
      IF(ACTNUM .EQ. 2) GO TO 999
      IF(ACTNUM .EQ. 3) GO TO 999
      IF(ACTNUM .EQ. 7) GO TO 999
      IF(ACTNUM .EQ. 10) GO TO 999
      IF(ACTNUM .EQ. 11) GO TO 999
      IF(ACTNUM .EQ. 12) GO TO 999
      IF(ACTNUM .EQ. 13) GO TO 999
      WRITE(10,900)TNOW
C  Write out current time
900   FORMAT(1X,'TIME ',F9.4)
      IF((ACTNUM .EQ. 4) .OR. (ACTNUM .EQ. 5)) THEN
C  Loader activity
        IF(ACTNUM .EQ. 4) THEN
C  Loader 1
          LOADTIME = ATRIB(4)/6.
          LOADNUM = 5
          WRITE(10,790)LOADNUM
C  Send loader to get load
790       FORMAT(1X,'PLACE ',I2,' ON Getload1')
        ENDIF
        IF(ACTNUM .EQ. 5) THEN
C  Loader 2
          LOADTIME = ATRIB(5)/6.
          LOADNUM = 6
          WRITE(10,795)LOADNUM
C  Send loader to get load
795       FORMAT(1X,'PLACE ',I2,' ON Getload2')
        ENDIF
        ATRIB(11) = LOADNUM
        WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
800     FORMAT(1X,'SET ',I2,' TRAVEL ',F9.4)
        WRITE(10,805)TNOW + LOADTIME
C  Increment time
805     FORMAT(1X,'TIME ',F9.4)
        WRITE(10,810)ATRIB(1) - 1
C  Place load on loader
810     FORMAT(1X,'DESTROY ',F3.0)
```

```
      WRITE(10,815)LOADNUM
C  Change loader to loaded
815   FORMAT(1X,'SET ',I2,' CLASS Loaded')
      WRITE(10,820)LOADNUM
C  Place load in truck
820   FORMAT(1X,'PLACE ',I2,' ON Dumpload')
      WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
      WRITE(10,805)TNOW + (3. * LOADTIME)
C  Increment time
      WRITE(10,825)LOADNUM
C  Change loader to unloaded
825   FORMAT(1X,'SET ',I2,' CLASS Loader')
      WRITE(10,830)ATRIB(10)
C  Change truck to partially loaded
830   FORMAT(1X,'SET ',F3.0,' CLASS Tload1')
      WRITE(10,835)LOADNUM
C  Send loader to get new load
835   FORMAT(1X,'PLACE ',I2,' ON Getnewload')
      WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
      WRITE(10,805)TNOW + (4. * LOADTIME)
C  Increment time
      WRITE(10,810)ATRIB(1)
C  Place load on loader
      WRITE(10,815)LOADNUM
C  Change loader to loaded
      WRITE(10,820)LOADNUM
C  Place load in truck
      WRITE(10,800)LOADNUM,LOADTIME
C  Set travel time
    ENDIF
    IF(ACTNUM .EQ. 6) THEN
C  Hauling activity
      LOADNUM = ATRIB(11)
      WRITE(10,825)LOADNUM
C  Change loader to unloaded
      IF(LOADNUM .EQ. 5) THEN
        WRITE(10,840)LOADNUM
C  Send loader 1 to rest spot
840     FORMAT(1X,'PLACE ',I2,' ON Return1')
      ENDIF
      IF(LOADNUM .EQ. 6) THEN
        WRITE(10,845)LOADNUM
C  Send loader 2 to rest spot
```

```
845       FORMAT(1X,'PLACE ',I2,' ON Return2')
          ENDIF
          WRITE(10,850)LOADNUM,ATRIB(6)
C  Set travel time
850       FORMAT(1X,'SET ',I2,' TRAVEL ',F3.0)
          WRITE(10,855)ATRIB(10)
C  Change truck to loaded
855       FORMAT(1X,'SET ',F3.0,' CLASS Tload2')
          WRITE(10,860)ATRIB(10)
C  Start truck hauling
860       FORMAT(1X,'PLACE ',F3.0,' ON Truckdump')
          WRITE(10,875)ATRIB(10),ATRIB(7)
C  Set travel time
        ENDIF
        IF(ACTNUM .EQ. 8) THEN
C  Dumping activity
          XX(1) = XX(1) + 1
          WRITE(10,861)ATRIB(10)
C  Start truck dumping
861       FORMAT(1X,'PLACE ',F3.0,' ON Dumping')
          WRITE(10,862)ATRIB(10),ATRIB(8)/2.
C  Set truck travel time
862       FORMAT(1X,'SET ',F3.0,' TRAVEL ',F9.4)
          WRITE(10,805)TNOW + (ATRIB(8)/2.)
C  Incrment time
          WRITE(10,865)ATRIB(10)
C  Change truck to unloaded
865       FORMAT(1X,'SET ',F3.0,' CLASS Truck')
          WRITE(10,866)XX(1)
C  Place two loads in truck
866       FORMAT(1X,'CREATE Twoloads ',F5.0)
          WRITE(10,867)XX(1)
C  Start two loads dumping
867       FORMAT(1X,'PLACE ',F5.0,' ON Loaddump')
        ENDIF
        IF(ACTNUM .EQ. 9) THEN
C  Return activity
          WRITE(10,870)ATRIB(10)
C  Start truck returning
870       FORMAT(1X,'PLACE ',F3.0,' ON Truckret')
          WRITE(10,875)ATRIB(10),ATRIB(9)
C  Set truck travel time
875       FORMAT(1X,'SET ',F3.0,' TRAVEL ',F9.4)
        ENDIF
        IT = 1
```

```
      RETURN

C  The following statements skip over nodes that are not
C  important for the animation.

100   IF(LABL .EQ. '    ') GO TO 999
      IF(LABL .EQ. 'LOAD') GO TO 999
      IF(LABL .EQ. 'LDER') GO TO 999
      IF(LABL .EQ. 'BEGN') GO TO 999
      IF(LABL .EQ. 'HAUL') GO TO 999
      IF(LABL .EQ. 'FRND') GO TO 999
      WRITE(10,900)TNOW
      IF(LABL .EQ. 'STAR') THEN
        II = II + 1
        WRITE(10,905)II
C  Create new load
905     FORMAT(1X,'CREATE Load ',I3)
        WRITE(10,910)II
C  Send load to load queue
910     FORMAT(1X,'PLACE ',I3,' ON Loadpath')
      ENDIF
      IF(LABL .EQ. 'TRKS') THEN
        WRITE(10,915)ATRIB(10)
C  Place truck in loading queue
915     FORMAT(1X,'PLACE ',F3.0,' ON Truckload')
        WRITE(10,920)ATRIB(10)
C  Set truck travel time
920     FORMAT(1X,'SET ',F3.0,' TRAVEL 2')
      ENDIF
999   IT = 1
      RETURN
      END
```

This is a sample of the programs used to created sorted Proof trace files. Only one is listed since the only differences in the programs are the input and output file names. This particular program took one unsorted trace file and created two sorted Proof trace files, one for M and one for MI. Thus, the trace files for M and MI were exactly the same. Also, the trace files for MC and MCI were the same. Differences in the animations were made by changing the animation layout files. All of the object classes for M and MC were changed to simple icons. For example, when the trace file changed the object class to loaded, no change was seed on the screen for M and MC because all the object classes were the simple icons. (See Figure 2 on page 17). Therefore, each scenario had to be run only five times instead of seven to created the seven Proof trace files.

```
      PROGRAM SORT
      INTEGER I,J,K,NUMLINES,LINE,FLAG,SORTING,READLINE
      CHARACTER*5 TIME
      CHARACTER*80 LINES(3500)
      REAL TIMES,TIMEARRY(3500,3),TEMP1,TEMP2,TEMP3
      OPEN(UNIT=10,FILE='ldermi.tr',STATUS='UNKNOWN')
      OPEN(UNIT=11,FILE='ldermi.atf',STATUS='UNKNOWN')
      OPEN(UNIT=12,FILE='lderm.atf',STATUS='UNKNOWN')
      I=0
      NUMLINES=0
      LINE=0
      FLAG=0
10    READ(10,FMT=100,END=200)TIME
100   FORMAT(A5)
      LINE = LINE + 1
      NUMLINES = NUMLINES + 1
      IF(TIME .EQ. ' TIME') THEN
        IF(FLAG .EQ. 0) THEN
          FLAG = 1
          NUMLINES = 0
          GO TO 10
        ENDIF
        I = I + 1
        TIMEARRY(I,1) = TIMES
        TIMEARRY(I,2) = NUMLINES - 1
        TIMEARRY(I,3) = LINE - NUMLINES
        BACKSPACE 10
        READ(10,FMT=101,ERR=195)TIMES
101     FORMAT(6X,F9.4)
        NUMLINES=0
      ENDIF
      GO TO 10
195   WRITE(11,110)
110   FORMAT(1X,'BACKSPACE ERROR')
200   I = I + 1
```

```
      TIMEARRY(I,1) = TIMES
      TIMEARRY(I,2) = NUMLINES - 1
      TIMEARRY(I,3) = LINE - NUMLINES
C
      REWIND 10
      READLINE = 1
15    READ(10,FMT=115,END=300)LINES(READLINE)
115   FORMAT(A60)
      READLINE = READLINE + 1
      GO TO 15
300   CONTINUE
C
      DO 25 RUNTHRU = 1,I
        DO 20 SORTING = 1,I-1
          IF(TIMEARRY(SORTING,1) .GT. TIMEARRY(SORTING+1,1)) THEN
            TEMP1 = TIMEARRY(SORTING+1,1)
            TIMEARRY(SORTING+1,1) = TIMEARRY(SORTING,1)
            TIMEARRY(SORTING,1) = TEMP1
            TEMP2 = TIMEARRY(SORTING+1,2)
            TIMEARRY(SORTING+1,2) = TIMEARRY(SORTING,2)
            TIMEARRY(SORTING,2) = TEMP2
            TEMP3 = TIMEARRY(SORTING+1,3)
            TIMEARRY(SORTING+1,3) = TIMEARRY(SORTING,3)
            TIMEARRY(SORTING,3) = TEMP3
          ENDIF
20      CONTINUE
25    CONTINUE
C
      DO 35 J = 1,I
        LINE = TIMEARRY(J,3)
        NUMLINES = TIMEARRY(J,2)
        DO 30 K = LINE,LINE + NUMLINES
          WRITE(11,115)LINES(K)
          WRITE(12,115)LINES(K)
30      CONTINUE
35    CONTINUE
      WRITE(11,125)
      WRITE(12,125)
125   FORMAT(1X,'END')
C
      CLOSE(10)
      CLOSE(11)
      CLOSE(12)
      END
```

Bibliography

Aiken, Peter and others. "Use of Multimedia to Augment Simulation," *Proceedings of the 1990 Winter Simulation Conference*. 775-783. New Orleans LA: IEEE Press, 1990.

Air Force Institute of Technology. *Introduction to AFIT Educational Computer Services*. Revision 5.3. Wright-Patterson AFB OH: AFIT, 22 May 1991.

Balci, Osman. "How to Assess the Acceptability and Credibility of Simulation Results," *Proceedings of the 1989 Winter Simulation Conference*. 62-71. Washington DC: IEEE Press, 1989.

----- and Robert G. Sargent. "A Bibliography on the Credibility Assessment and Validation of Simulation and Mathematical Models," *Simuletter, 15*: 15-27 (July 1984).

Banks, Jerry. "Testing, Understanding, and Validating Complex Simulation Models," *Proceedings of the 1989 Winter Simulation Conference*. 549-551. Washington DC: IEEE Press, 1989.

Brunner, Daniel T. and others. "Proof Animation: The General Purpose Animator," *Proceedings of the 1991 Winter Simulation Conference*. 90-94. Atlanta GA: IEEE Press, 1991.

Budescu, David V. and others. "A Comparison of the Eigenvalue Method and the Geometric Mean Procedure for Ratio Scaling," *Applied Psychological Measurement,* 10: 69-78 (March 1986).

Carson, John S. "Verification and Validation: A Consultant's Perspective," *Proceedings of the 1989 Winter Simulation Conference*. 552-558. Washington DC: IEEE Press, 1989.

----- and Onala M. Atala. "Using Computer Simulation for Rapid Transit Operating Strategies," *Proceedings of the 1990 Winter Simulation Conference*. 798-801. New Orleans LA: IEEE Press, 1990.

Cyr, Rodney W. "Using Animation to Enhance a Marine-Terminal Monte Carlo Simulator," *Proceedings of the 1992 Winter Simulation Conference*. 1000-1003. Arlington VA: IEEE Press, 1992.

Hollocks, Brian W. "Practical Benefits of Animated Graphics in Simulation," *Proceedings of the 1984 Winter Simulation Conference*. 323-328. Dallas TX: IEEE Press, 1984.

Johnson, M. Eric and Jacob P. Poorte. "A Hierarchical Approach to Computer Animation in Simulation Modeling," *Simulation, 50*: 30-36 (January 1988).

Kalasky, David R. and Deborah A. Davis. "Computer Animation With CINEMA," *Proceedings of the 1991 Winter Simulation Conference*. 122-127. Atlanta GA. IEEE Press, 1991.

Law, Averill M. and W. David Kelton. *Simulation Modeling and Analysis*. New York: McGraw-Hill, 1991.

Neter, John and others. *Applied Linear Statistical Models* (Third Edition). Boston MA: Richard D. Irwin, Inc., 1990.

Pritsker, A. Alan B. *Introduction to Simulation and SLAM II* (Third Edition). West Lafayette IN: Systems Publishing Corporation, 1986.

Saaty, Thomas L. *The Analytic Hierarchy Process.* New York: McGraw-Hill, 1980.

Sargent, Robert G. "Simulation Model Verification and Validation," *Proceedings of the 1991 Winter Simulation Conference.* 37-47. Atlanta GA. IEEE Press, 1991.

-----. "The Use of Graphical Models in Model Validation," *Proceedings of the 1986 Winter Simulation Conference.* 237-241. Washington DC. IEEE Press 1986.

SAS Institute Inc. *SAS/STAT User's Guide, Version 6, Fourth Edition, Volume 1,* Cary NC: SAS Institue Inc., 1989.

Schlesinger, Stewart and others. "Terminology for Model Credibility," *Simulation, 32*: 103-104 (March 1979).

Schuppe, Thomas F. "Modeling and Simulation: A Department of Defense Critical Technology," *Proceedings of the 1991 Winter Simulation Conference.* 519-525. Atlanta GA. IEEE Press, 1991.

Standridge, Charles R. "Animating Simulations Using TESS," *Computers and Industrial Engineering, 10*: 121-134 (1986).

SWORD. Version 1.0, IBM. Computer software. Harry G. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson AFB OH, 1989.

Vidulich, Michael A. "The Use of Judgment Matrices in Subjective Workload Assessment: The Subjective WORkload Dominance (SWORD) Technique," *Proceedings of the Human Factors Society 33rd Annual Meeting.* 1406-1410. Santa Monica CA. Human Factors Society, 1989.

-----. Armstrong Laboratory, Wright-Patterson Air Force Base OH. Telephone interview. November 1992.

----- and Pamela S. Tsang. "Absolute Magnitude Estimation and Relative Judgment Approaches to Subjective Workload Assessment," *Proceedings of the Human Factors Society 31st Annual Meeting.* 1057-1061. Santa Monica CA. Human Factors Society, 1987.

Whitner, Richard B. and Osman Balci. "Guidelines for Selecting and Using Simulation Model Verification Techniques," *Proceedings of the 1989 Winter Simulation Conference.* 559-568. Washington DC: IEEE Press, 1989.

Williams, Cindy and Gordon Crawford. *Analysis of Subjective Judgment Matrices.* Technical Report R-2572-AF. Santa Monica CA: RAND, May 1980.

-----. *Analysis of Subjective Judgment Matrices.* Technical Report R-2572-1-AF. Santa Monica CA: RAND, May 1985.

Vita

Captain Michael L. Carpenter was born on 14 November 1960 in Hattiesburg, Mississippi. He graduated from Hattiesburg High School in 1978 and attended the University of Southern Mississippi, graduating with a Bachelor of Science in Mathematics in May 1982. Upon graduation, he received a reserve commission in the USAF and served his first tour of duty at the Tactical Fighter Weapons Center, Nellis AFB, Nevada. He served as an analyst and project manager for the 57th Fighter Weapons Wing Tactics and Test. In January of 1988, he was assigned to the Joint Data Systems Support Center, Defense Communications Agency, at the Pentagon. While at the Pentagon, he attended Squadron Officer School in residence. He entered the School of Engineering, Air Force Institute of Technology, in August of 1991, pursuing the degree of Master of Science in Operations Research.

Permanent address: 307 Venetian Way
Hattiesburg, Mississippi
39401

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
March 1993

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE
USING ANIMATION IN THE VALIDATION OF SIMULATION MODELS

5. FUNDING NUMBERS

6. AUTHOR(S)
Michael L. Carpenter, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GOR/ENS/93M-02

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
HQ AFOTEC/SAL
Kirtland AFB NM 87117

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This study examined three aspects of animation (movement, color, and detail of icons) to determine which one (or ones) best communicated the operation of a simulation model. The procedure was done in the context of using animation to establish a model's face validity. Movement, color, and detail of icons were looked at individually and in combination. The ability to communicate was measured both subjectively and objectively. The subjective measures were a selection of "best" and "worst" animation types where "best" and "worst" referred to how well an animation communicated, and a pairwise comparison of the animation types which resulted in preference ratings for each animation. There were seven different scenarios containing various problems with the system. The objective measures were subject problem identification accuracy and time delay of problem identification. The results showed that movement in animations was always preferred to a lack of movement in animations. However, movement, color, and detail of icons in combination was preferred the most. Objectively, movement was the most important aspect. The subjects performed equally well for all the animations with movement and, when there was no movement, performance dropped.

14. SUBJECT TERMS
Computerized simulation, Simulation, Operations Research, Animation, Validation

15. NUMBER OF PAGES
170

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements***.

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	- Contract	PR	- Project
G	- Grant	TA	- Task
PE	- Program Element	WU	- Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

- **DOD** - See DoDD 5230.24, "Distribution Statements on Technical Documents."
- **DOE** - See authorities.
- **NASA** - See Handbook NHB 2200.2.
- **NTIS** - Leave blank.

Block 12b. Distribution Code.

- **DOD** - Leave blank.
- **DOE** - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
- **NASA** - Leave blank.
- **NTIS** - Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

END

FILMED

DATE: 4-93

DTIC